

Manual comunitario de ayuda HP Prime

Contenido completo, comandos, funciones y jerarquía de navegación

Documento no oficial para fines de referencia técnica y comunitaria

Acerca de esta edición

Este documento reorganiza la ayuda integrada de HP Prime en un formato de lectura continua y navegable. La jerarquía sigue el árbol que utiliza la interfaz del HP Prime Virtual Calculator de la versión analizada. El contenido descriptivo procede del recurso español HelpEs.hpresource; cuando una entrada carece de título localizado, se conserva el identificador técnico de HelpEn.hpresource y se acompaña de una explicación española obtenida del propio contenido de ayuda en español.

Los nombres ejecutables de funciones, variables y métodos se conservan sin traducir para que puedan escribirse exactamente en la calculadora. La traducción se aplica a su rótulo o explicación, no al identificador que forma parte de la sintaxis.

Criterio de traducción

Ejemplo: BondYield se conserva como nombre de función y se acompaña de su explicación española. Así se evita sustituir un identificador ejecutable por una palabra que la calculadora no reconocería.

HP Prime y las denominaciones de sus productos pertenecen a sus respectivos titulares. Esta es una edición comunitaria no oficial.

Resumen de cobertura

2,351	2,324	3,620
entradas de ayuda	IDs visibles únicos	apariciones en el árbol

Las apariciones superan a los IDs únicos porque una misma ayuda puede enlazarse desde varias ramas. Las 27 entradas que no pertenecen al árbol visible se conservan en un apéndice.

Índice general

Acerca de la HP Prime

Gracias

Vista de Inicio

Configuración de Inicio

Guardar y cargar en la Vista de Inicio

Sistema algebraico computacional (CAS)

Vista CAS

Config. sist. algebraico comp.

Aplicaciones del sistema

Vistas de las aplicaciones principales

Variables de aplicaciones comunes

Variables comunes del modo aplicaciones

Aplicación Función

Aplicación Gráficos avanzados

Aplicación Gráficos 3D

Aplicación Geometría

Aplicación Hoja de cálculo

Aplicación Var 1 estadística

Aplicación Var 2 estadística

Aplicación Inferencia

Aplicación Data Streamer

Aplicación Solucionador

Aplicación Solucionador lineal

Aplicación Solucionador de triángulo

Aplicación Finanzas

Aplicación Explorador

Aplicación Paramétrica

Aplicación Secuencia

Aplicación Polar

Aplicación Python

Índice de comandos de Inicio

Funciones en la vista de Inicio (PPL): 'A'

Funciones en la vista de Inicio (PPL): 'B'

Funciones en la vista de Inicio (PPL): 'C'

Funciones en la vista de Inicio (PPL): 'D'

Funciones en la vista de Inicio (PPL): 'E'

Funciones en la vista de Inicio (PPL): 'F'

Funciones en la vista de Inicio (PPL): 'G'

Funciones en la vista de Inicio (PPL): 'H'

Funciones en la vista de Inicio (PPL): 'I'

Funciones en la vista de Inicio (PPL): 'K'

Funciones en la vista de Inicio (PPL): 'L'

Funciones en la vista de Inicio (PPL): 'M'

Funciones en la vista de Inicio (PPL): 'N'

Funciones en la vista de Inicio (PPL): 'O'

Funciones en la vista de Inicio (PPL): 'P'

Funciones en la vista de Inicio (PPL): 'Q'

Funciones en la vista de Inicio (PPL): 'R'

Funciones en la vista de Inicio (PPL): 'S'

Funciones en la vista de Inicio (PPL): 'T'

Funciones en la vista de Inicio (PPL): 'U'

Funciones en la vista de Inicio (PPL): 'V'

Funciones en la vista de Inicio (PPL): 'W'

Funciones en la vista de Inicio (PPL): 'X'

Funciones en la vista de Inicio: símbolos

Índice de comandos CAS

Funciones en vista CAS: 'A'

Funciones en vista CAS: 'B'

Funciones en vista CAS: 'C'

Funciones en vista CAS: 'D'

Funciones en vista CAS: 'E'
Funciones en vista CAS: 'F'
Funciones en vista CAS: 'G'
Funciones en vista CAS: 'H'
Funciones en vista CAS: 'I'
Funciones en vista CAS: 'J'
Funciones en vista CAS: 'K'
Funciones en vista CAS: 'L'
Funciones en vista CAS: 'M'
Funciones en vista CAS: 'N'
Funciones en vista CAS: 'O'
Funciones en vista CAS: 'P'
Funciones en vista CAS: 'Q'
Funciones en vista CAS: 'R'
Funciones en vista CAS: 'S'
Funciones en vista CAS: 'T'
Funciones en vista CAS: 'U'
Funciones en vista CAS: 'V'
Funciones en vista CAS: 'W'
Funciones en vista CAS: 'X'
Funciones en vista CAS: 'Z'
Funciones en vista CAS: símbolos

Índice de comandos Python

Funciones en Python: 'A'
Funciones en Python: 'B'
Funciones en Python: 'C'
Funciones en Python: 'D'
Funciones en Python: 'E'
Funciones en Python: 'F'
Funciones en Python: 'G'
Funciones en Python: 'H'
Funciones en Python: 'I'
Funciones en Python: 'J'
Funciones en Python: 'K'
Funciones en Python: 'L'
Funciones en Python: 'M'
Funciones en Python: 'N'
Funciones en Python: 'O'
Funciones en Python: 'P'
Funciones en Python: 'Q'
Funciones en Python: 'R'
Funciones en Python: 'S'
Funciones en Python: 'T'
Funciones en Python: 'U'
Funciones en Python: 'V'
Funciones en Python: 'W'
Funciones en Python: 'Y'
Funciones en Python: 'Z'
Operadores
Excepciones

Menús del cuadro de herramientas

Menú Matem.
Menú CAS
Menú Apl.
Menú Usua.
Menú Catlg.

Unidades

Herramientas de unidad
Unidades
Constantes

Variables

Variables de Inicio
Variables del sistema algebraico computacional (CAS, por sus siglas en inglés)
Variables de aplicación

Variables de usuario

Catálogo Vars

Variables de PPL

Gestor de memoria

Copias de seguridad

Menú de caracteres

Paletas de atajo

Plantilla de matemáticas

Paleta de caracteres especiales

Paleta de relaciones

Teclado

Conversiones Decimales: a_b/c

Almohadilla direccional

Tecla X,T,θ,N

Retroceso

Tangente

Función de cuadrado

Función de potencia

Dividir

Coma

Numerales

Multiplicar

Restar

Agreg.

Punto decimal

Tecla Enter

Crear una función definida por el usuario

Grados Minutos Segundos

Tecla EEX

Tecla Eliminar

Función exponencial

Raíz cuadrada

Copiar

Pegar

Inverso

Ángulo

Tecla Base

Tecla igual

Tecla CAS

Tecla Vars

Tecla Apps

Vista simbólica

Vista de gráfico

Vista numérica

Tecla de menú

Tecla Escape

Paréntesis

Tecla ALPHA

Tecla Shift

Tecla On

Tecla espaciadora

Tecla Sto►

Vista Config. simbólica

Vista de Configuración de gráfico

Vista Configuración numérica

Comillas simples

Llaves

Corchetes

Tecla Eval

Paleta de caracteres especiales

Paleta de relaciones

Guión bajo

Config. sist. algebraico comp.

Tecla Del

Tecla Help (Ayuda)

Teclado del usuario

≈

Valor de absoluto

Coseno inverso

Última respuesta

Seno inverso

Tangente inversa

Función coseno

Logaritmo natural

Logaritmo general

Número negativo

Tecla de raíz n-ésima

Seno

Plantilla de matemáticas

Caracteres matemáticos

Aritmética de enteros

Cambiar la base de números enteros

Manipular números enteros

Catálogo de lista

Editor de lista

Catálogo de matriz

Editor de matriz

Catálogo de programas

Editor de programa

Catálogo de notas

Editor de notas

Modo Examen

Configuración de modo Examen

Creación de configuraciones nuevas

Activación del Modo Examen

Mensajería

Encuesta y cuestionario

Ayuda principal

ID 1 · título localizado en HelpEs

Este es el sistema de ayuda en línea para su HP Prime.

En todo el sistema de ayuda, la palabra "presionar" siempre se refiere a una tecla física en el teclado Prime. La palabra "pulsar" siempre se refiere a un objeto que se muestra en la pantalla táctil, con frecuencia un elemento en el menú del software (un "botón" de menú) en la parte inferior de la pantalla.

Presione la parte superior o inferior de la rueda, arrastre o deslícese por la pantalla táctil, o pulse cualquiera de los lados de la opción [-▲ Página ▼-] para desplazarse por cualquier página de ayuda (como ésta). Para salir de una página de ayuda, pulse [[OK]] o Esc.

Para ver todos los temas de ayuda, pulse [[Árbol]].

- Navegue utilizando la rueda, arrastre o deslícese.
- Toque [+] o [-] o presione la rueda hacia la derecha o izquierda para expandir/contrair un grupo de temas.
- Presione Enter u [[OK]] para ver un tema.

Pulse [[Teclas]] y, a continuación, presione cualquier tecla del teclado para ver la ayuda relacionada con ella.

Pulse [[Buscar]], ingrese texto de búsqueda (mantenga pulsada la tecla ALPHA para ingresar caracteres alfabéticos) y presione Enter o pulse [[OK]]. Si "Buscar en contenido" no está activado, solo se buscará en los títulos de la ayuda.

Como regla general, al pulsar Help aparecerá la página de ayuda más relevante para la pantalla o selección actual. Por ejemplo, al pulsar Help cuando la Biblioteca de aplicaciones está abierta, aparecerá una descripción general de la aplicación seleccionada en ese momento. Una vez abierta una aplicación, hay una página de ayuda diferente para cada vista de la aplicación (Simbólica, Gráfico, Numérico, etc.) que explicará el objetivo de la vista y los elementos de menú que aparecen en la parte inferior de la pantalla.

0002 Acerca de la HP Prime

ID 2 · título localizado en HelpEs

Calculadora gráfica HP Prime

[versión]

© [fecha] HP Development Company, L.P.

Para ver la información de derechos de autor y licencia de un componente de software, toque el menú [[Otro]], seleccione un componente y presione la tecla Enter.

0003 Gracias

ID 3 · título localizado en HelpEs

Jeri, Teddy, Mark, Ruth, Dave

Bernard, Conrad, Cyrille, Gerald, GT, Jean-Yves, Jeff, Louis, Matt, Rian, Tim

Angie, Bill, Chris, Craig, Glenn, Jason, Jason, Julia, Mike, Michelle, Shan-shan, Syed, Geoff, Charlotte, George, Jack, James, Katy, Marcia, Mike, Tyler, Anusha, Archana, Dean, Piotr

Eddie, Edwin, Erik, Fabien, Felix, Gilles, Han, Joe, Johnny, Katie, Marv, Namir, Natasha, Patrice, Stefan, Wes

0005 Vista de Inicio

ID 5 · título localizado en HelpEs

La pantalla básica de la calculadora es la Vista de Inicio. La mayoría de los cálculos se realizan aquí. Ingrese una expresión en el mismo orden de izquierda a derecha en la cual la escribiría.

También puede hacer cálculos utilizando la Notación polaca inversa (RPN) si ha seleccionado RPN como su opción preferida de entrada en la Configuración de Inicio.

Una expresión puede contener números, llamadas a funciones, variables, listas y matrices. Para introducir una expresión, presione las teclas correspondientes o seleccione elementos de uno de los menús, como el menú Caja de herramientas. También puede introducir el nombre de una función utilizando la tecla ALPHA para seleccionar el modo de entrada alfabética adecuado. Una vez que haya terminado de introducir la expresión, presione Enter para evaluarla.

Puede guardar una respuesta asignándosela a una variable que luego puede usar en cálculos posteriores.

Después de la evaluación, tanto la expresión como su resultado se muestran en la sección de historial de la Vista de Inicio. Puede desplazarse por el historial utilizando la rueda, deslizando o arrastrando el dedo. Puede volver a usar una expresión o resultado pulsando en él dos veces. Se copia en la línea de entrada y queda lista para su edición. (También puede desplazarse a un elemento del historial y pulsar `[[Copiar]]`).

Los elementos del menú de botón táctil en la Vista de Inicio son:

- `[[Almc ►]]` permite almacenar datos en una variable
- `[[Copiar]]` permite copiar la entrada o el resultado seleccionados en la línea de entrada, en la posición actual del cursor
- `[[Mostr.]]` muestra el elemento seleccionado en modo de pantalla completa (con desplazamiento horizontal y vertical habilitado)

Además del botón `[[Copiar]]`, que solo funciona en la Vista de inicio y la Vista CAS, puede utilizar el portapapeles para copiar y pegar expresiones. Sus últimas entradas se copian automáticamente en el portapapeles. Presione Shift Menu (Paste) para abrir el portapapeles. Utilice la rueda para seleccionar una expresión del portapapeles y, a continuación, pulse `[[OK]]` o presione Esc. Para agregar manualmente un elemento al portapapeles, selecciónelo y presione Shift View (Copy).

Presione la tecla de Retroceso para eliminar el carácter a la izquierda del cursor en la línea de entrada. Para eliminar toda la línea, presione Esc. Para eliminar todo el historial, presione Shift Esc (Clear).

0006 Configuración de Inicio

ID 6 · título localizado en HelpEs

La página 1 de la Configuración de Inicio contiene las siguientes opciones:

- Medida del ángulo: seleccione Radianes, Grados o Gradianes
- Formato de número: seleccione Estándar, Fijo, Científico, Ingeniería, Flotante o Redondeado
- Agrupación de dígitos: seleccione el formato de agrupamiento de dígitos.
- Entrada: permite seleccionar Algebraica, RPN o Libro de texto
- Enteros: seleccione la base predeterminada para aritmética de enteros: Binaria, Octal, Decimal o Hex. También puede elegir el tamaño de la palabra (o la cantidad de bits) y si los enteros son con o sin signo.
- Complejo: permite seleccionar un formato para representar números complejos y permite resultados de números complejos a partir de datos reales
- Idioma: permite seleccionar un idioma

Si aparece "Ver Configuración simbólica" en lugar del campo de entrada de un ajuste, la aplicación actual sustituye la configuración actual. Para cambiar la configuración, hágalo desde la Configuración simbólica. Si desea borrar la anulación específica de la aplicación y volver al valor de todo el sistema de la configuración, vaya a Configuración simbólica y cambie el valor de la configuración a "Sistema".

0007 Página 1

ID 7 · título localizado en HelpEs

La página 1 de la Configuración de Inicio contiene las siguientes opciones:

- Medida del ángulo: seleccione Radianes, Grados o Gradianes
- Formato del número: seleccione Estándar, Fijo, Científico, Ingeniería, Flotante o Redondeado
- Agrupación de dígitos: seleccione el formato de agrupamiento de dígitos.
- Entrada: permite seleccionar Algebraica, RPN o Libro de texto
- Enteros: seleccione la base predeterminada para aritmética de enteros: Binaria, Octal, Decimal o Hex. También puede elegir el tamaño de la palabra (o la cantidad de bits) y si los enteros son con o sin signo.
- Complejo: permite seleccionar un formato para representar números complejos y permite resultados de números complejos a partir de datos reales
- Idioma: permite seleccionar un idioma

Si aparece "Ver Configuración simbólica" en lugar del campo de entrada de un ajuste, la aplicación actual sustituye la configuración actual. Para cambiar la configuración, hágalo desde la Configuración simbólica. Si desea borrar la anulación específica de la aplicación y volver al valor de todo el sistema de la configuración, vaya a Configuración simbólica y cambie el valor de la configuración a "Sistema".

0009 Formatos de número

ID 9 · título localizado en HelpEs

El HP Prime le ofrece varias opciones de formato de números para controlar la visualización de los resultados numéricos en la vista Home.

- Estándar Muestra resultados con precisión completa, mostrando todos los dígitos significativos del cálculo.
- Fijo n Muestra resultados redondeados a n decimales.
- Científico n Muestra resultados en notación científica con n+1 dígitos significativos.
- Ingeniería n Similar a "Científico n", pero el exponente siempre es un múltiplo de 3.
- Flotante n Muestra resultados con hasta n+1 dígitos significativos, omitiendo ceros finales.
- Redondeado n Muestra resultados redondeados a n decimales, omitiendo ceros finales.

0008 Métodos de entrada

ID 8 · título localizado en HelpEs

La HP Prime ofrece tres maneras de ingresar expresiones en la Vista de Inicio.

- Libro de texto Una expresión se introduce del mismo modo que si escribiera en un papel. Por ejemplo, una división estará representada por una barra de división con el dividendo por encima y por debajo del divisor.
- Algebraico Una expresión se introduce en una sola línea de texto.
- RPN (Notación polaca inversa /Reverse Polish Notation/) Los argumentos de la expresión se introducen primero, seguidos por el operador. La entrada de un operador evalúa automáticamente lo que ya se ha introducido.

Ejemplo

Supongamos que quiere calcular $2 + 3$

En los modos algebraicos y libro de texto, es necesario introducir $2 + 3$, Enter

En el modo RPN, se introduce 2, Enter, 3 +

Si posteriormente introdujo $12 +$, 12 se suma automáticamente a la última respuesta en el modo RPN.

0010 Página 2

ID 10 · título localizado en HelpEs

La página 2 de la Configuración de Inicio contiene las siguientes opciones:

- Tamaño de la fuente: seleccione el tamaño de la fuente Pequeño, Mediano o Grande para la mayoría del texto mostrado.
- Nombre de la calculadora: brinde a su HP Prime un nombre particular. Esto contribuye con el reconocimiento en la comunicación inalámbrica con el Kit de conectividad
- Pant. Libro de texto: permite activar o desactivar la visualización del libro de texto.
- Pantalla del menú: si se selecciona, las funciones matemáticas y los comandos se representan en menús mediante un nombre descriptivo. Si no se selecciona, se representan mediante el nombre del comando.
- Matemáticas inteligentes: si se selecciona, los cálculos sobre números reales se presentan en forma exacta utilizando la aproximación real a cociente.

- [a b/c Opciones]: permite seleccionar las formas permitidas para mostrar resultados numéricos en forma exacta; al presionar [a b/c], se alterna entre la representación en coma flotante y las formas seleccionadas cuando están disponibles.
- Hora: permite configurar la hora y el formato de la hora. Permite activar o desactivar el reloj en la barra de título.
- Fecha: permite configurar la fecha y el formato de la fecha.
- Tema del color: permite seleccionar Claro u Oscuro, además de un color de subrayado.

0011 Página 3

ID 11 · título localizado en HelpEs

Todas las opciones de la página 3 de la Configuración de Inicio están relacionadas con el modo Examen. Hay un tema de ayuda independiente para el modo Examen.

0012 Página 4

ID 12 · título localizado en HelpEs

Si su HP Prime es compatible con el módulo inalámbrico, las opciones de la página 4 le permiten conectarse a una red inalámbrica de HP Classroom mediante dicho módulo y el Kit de conectividad HP. Para que aparezcan estas opciones, el adaptador inalámbrico debe estar conectado.

Para establecer la conexión, seleccione el nombre de la red en la lista desplegable situada en la parte superior del cuadro de diálogo. Las redes detectadas se añadirán automáticamente.

Si utiliza la aplicación HP Prime Pro, puede conectarse a una red IP de HP Classroom mediante la conexión por cable o inalámbrica de su dispositivo.

En primer lugar, asegúrese de que su dispositivo se encuentre en la misma red y subred que el ordenador en el que se ejecuta el Kit de conectividad HP. A continuación, compruebe que el Kit de conectividad HP esté configurado correctamente en el cuadro de diálogo de preferencias. En algunos ordenadores, el cortafuegos o el software antivirus pueden bloquear la conexión.

Para establecer la conexión, seleccione el nombre de la red en la lista desplegable situada en la parte superior del cuadro de diálogo. Las redes IP de HP Classroom detectadas se añadirán automáticamente a la lista, por encima de la entrada «Add connection...».

Para conectarse a un ordenador que no aparece en la lista, seleccione «Add connection...» e introduzca el nombre de host o la dirección IP del ordenador en el que se ejecuta el Kit de conectividad HP. También puede proporcionar un nombre descriptivo para la conexión, que se mostrará en la lista de conexiones de red, debajo de la entrada «Add connection...». Si deja en blanco el campo del nombre descriptivo, se mostrará en la lista el nombre de host o la dirección IP. Una vez añadida la conexión, su HP Prime intentará conectarse.

También puede escribir directamente en el cuadro de la lista desplegable un nuevo nombre de red con el formato Nombre_de_host_o_IP@Nombre_descriptivo. La parte @Nombre_descriptivo es opcional.

El HP Prime solo recuerda las seis últimas redes añadidas. La conexión utilizada más recientemente aparece en la parte superior de la lista de redes añadidas. Mientras la lista desplegable esté abierta, seleccione una red y pulse la tecla Retroceso para eliminarla de la lista. Toque la tecla de menú [[Editar]] para modificar la conexión.

1743 Guardar y cargar en la Vista de Inicio

ID 1743 · título localizado en HelpEs

El cuadro de diálogo de guardar y cargar en la Vista de Inicio le permite guardar cálculos recientes en un archivo y volver a cargarlos más tarde.

Esta es una función de "guardar su trabajo" que le permite guardar en qué está trabajando y volver a ello más tarde.

Presione el botón [[Guard.]] para guardar el estado actual. Puede seleccionar lo que desea guardar: cálculos, variables, listas y matrices.

Ingrese un nombre para su copia de seguridad o seleccione uno existente (que será sobrescrito).

Presione el botón [[Cargar]] para cargar un estado anterior. Nuevamente, puede seleccionar lo que desea cargar desde el archivo. Presione [[Aceptar]] para completar la operación (tenga en cuenta que al cargar, los elementos actuales serán sobrescritos por los cargados).

0013 Sistema algebraico computacional (CAS)

ID 13 · título localizado en HelpEs

El Sistema algebraico computacional (CAS) permite realizar cálculos simbólicos.

De forma predeterminada, el Sistema algebraico computacional (CAS) trabaja en el modo exacto, y ofrece resultados aritméticos simbólicos o exactos. Por otro lado, los cálculos que no se realizan en CAS, como los realizados en la Vista de inicio o por una aplicación, son cálculos numéricos y, a menudo, son aproximaciones limitadas por la precisión de la calculadora.

Por ejemplo, $1/3 + 2/7$ da la respuesta aproximada de 0,619047619047 en la vista de Inicio (con formato numérico estándar); sin embargo, en el sistema algebraico computacional la respuesta es $13/21$.

El sistema algebraico computacional ofrece cientos de funciones, entre ellas, álgebra, cálculos, cálculos de ecuaciones, polinomios y más. Puede seleccionar una función

del menú Sistema algebraico computacional, uno de los menús del cuadro de herramientas. También puede seleccionar una función CAS en el menú Catlg (otro de los menús del cuadro de herramientas).

0014 Vista CAS

ID 14 · título localizado en HelpEs

Los cálculos CAS se realizan en la Vista CAS.

La vista del sistema algebraico computacional es casi idéntica a la vista de Inicio. Se genera un historial de cálculos y puede seleccionar y copiar cálculos previos de la misma forma que puede hacerlo en la vista de Inicio, así como almacenar objetos en variables.

Para abrir la vista del sistema algebraico computacional, pulse CAS. Aparece la etiqueta "Sistema algebraico computacional" en la parte izquierda de la barra de título para indicar que se encuentra en la vista del sistema algebraico computacional y no en la vista de Inicio.

Los botones del menú de la vista del sistema algebraico computacional son:

- [[Almc ►]] asigna un objeto a una variable
- [[Simplif.]] aplica reglas de simplificación comunes para reducir una expresión a su forma más simple
- [[Copiar]] permite copiar una entrada seleccionada en el historial a la línea de entrada.
- [[Mostr.]] muestra el elemento seleccionado en modo de pantalla completa (con desplazamiento horizontal y vertical habilitado)
- [[=]] Inserta un signo de igualdad (=).

0015 Config. sist. algebraico comp.

ID 15 · título localizado en HelpEs

Varios ajustes le permiten configurar cómo funciona el CAS. Para mostrar los ajustes, presione Shift + CAS. Las opciones están repartidas en dos páginas.

0016 Página 1

ID 16 · título localizado en HelpEs

Las opciones en la página 1 de la configuración CAS son:

- Medición de ángulo: seleccione las unidades para las mediciones de ángulos: radianes, grados o gradianes.
- Formato numérico (primera lista desplegable): seleccione el formato de número para las soluciones visualizadas: Estándar, Científico o Ingeniería.
- Formato numérico (segunda lista desplegable): Permite seleccionar el número de dígitos que se mostrará en modo aproximado (mantisa + exponente).
- Enteros (lista desplegable): seleccione la base del número entero: decimal (base 10), hexadecimal (base 16) u octal (base 8).

- **Enteros (casilla de verificación):** Si está activada, cualquier número real equivalente a un entero en un entorno sin sistema algebraico computacional se convertirá a entero en este sistema. (Los números reales no equivalentes a enteros se tratan como números reales en el sistema algebraico computacional tanto si esta opción está activada como si no).
- **Simplificar:** seleccione Ninguna simplificación automática o Mínima o Máxima.
- **Exacta:** si se selecciona, el CAS se encuentra en el modo exacto y las soluciones serán simbólicas. Si no se selecciona, el CAS está en modo aproximado y las soluciones serán aproximadas. Por ejemplo, $26/5$ devuelve $26/5$ en modo exacto y 5.2 en modo aproximado.
- **Usar $\sqrt{}$:** Si está activada, los polinomios se factorizan usando raíces cuadradas.
- **Principal:** Si está activada, se mostrarán las soluciones principales para las funciones trigonométricas. Si no está activada, se mostrarán las soluciones generales para las funciones trigonométricas.
- **Creciente:** Si está activada, los polinomios se mostrarán con potencias crecientes (por ejemplo, $-4+x+3x^2+x^3$). Si no está activada, los polinomios se mostrarán con potencias decrecientes (por ejemplo, x^3+3x^2+x-4).

0017 Página 2

ID 17 · título localizado en HelpEs

Las opciones en la página 2 de la configuración CAS son:

- **Evaluación recursiva** Permite especificar el número máximo de variables integradas permitidas en una evaluación interactiva. Consulte también Sustitución recursiva más abajo.
- **Sustitución recursiva** Permite especificar el número máximo de variables integradas permitidas en una sola evaluación en un programa. Consulte también Evaluación recursiva más arriba.
- **Función recursiva** Permite especificar el número máximo de llamadas de función integradas permitidas.
- **Epsilon** Cualquier número inferior al valor especificado para ϵ se mostrará como cero en algunos algoritmos numéricos.
- **Probabilidad** Permite especificar la probabilidad máxima de error de una respuesta para algoritmos no deterministas. Configure este valor como cero para algoritmos deterministas.
- **Newton** Permite especificar el número máximo de iteraciones al utilizar el método de Newton para buscar las raíces de un polinomio.

0018 Aplicaciones del sistema

ID 18 · título localizado en HelpEs

Gran parte de la funcionalidad de la calculadora HP Prime se ofrece a través de paquetes denominados aplicaciones de HP. La HP Prime incluye una colección de aplicaciones incorporadas para la representación gráfica, la geometría, la estadística, la inferencia, la resolución de ecuaciones, los cálculos financieros, la recopilación de datos, la programación y otras tareas matemáticas.

Para abrir una aplicación, pulse la tecla Apps para mostrar la Biblioteca de aplicaciones y, a continuación, toque el icono de la aplicación que desee utilizar. También puede resaltar una aplicación utilizando la rueda y pulsar Enter o tocar [[Inicio]].

Puede crear copias de las aplicaciones que admiten personalización. Cada copia almacena su propia configuración y sus propios datos, lo que le permite conservar distintas configuraciones para diferentes tareas o personalizar una aplicación mediante un programa asociado. Las aplicaciones personalizadas aparecen en la Biblioteca de aplicaciones junto con las aplicaciones incorporadas y se abren de la misma manera.

Para crear una copia de una aplicación:

- Pulse Apps para abrir la Biblioteca de aplicaciones.
- Resalte la aplicación que desee utilizar como base para la nueva aplicación, pero no la abra.
- Toque [[Guard]].
- Escriba un nombre para la nueva aplicación.
- Toque [[OK]] para confirmar el nombre y completar la operación.

La nueva aplicación aparecerá en la Biblioteca de aplicaciones. Puede abrirla de la misma manera que cualquier otra aplicación.

Las aplicaciones incorporadas no se pueden eliminar. Las copias que cree se pueden eliminar de la Biblioteca de aplicaciones.

0019 Vistas de las aplicaciones principales

ID 19 · título localizado en HelpEs

La mayoría de las aplicaciones tienen tres vistas principales: la simbólica, de gráfico y la numérica. Estas vistas se basan en las representaciones simbólica, gráfica y numérica de los objetos matemáticos. Se accede a ellas mediante las teclas Symb, Plot y Num cerca del área superior izquierda del teclado. Por lo general, estas vistas le permiten definir un objeto matemático (como una expresión o una sentencia abierta), trazarlo y ver los valores que genera.

Cada una de estas vistas va acompañada de una vista de configuración que permite configurar la apariencia de los datos en la vista principal correspondiente. Estas vistas se denominan Config. simbólica, Config. de gráfico y Configuración numérica. Puede acceder a ellas pulsando Shift Symb, Shift Plot y Shift Num respectivamente.

No todas las aplicaciones tienen las seis vistas descritas. El alcance y la complejidad de cada aplicación determina su conjunto particular de vistas. Por ejemplo, la aplicación Hoja de cálculo solo tiene una Vista numérica y el Explorador cuadrático solo tiene una Vista de gráfico.

0020 Vista simbólica

ID 20 · título localizado en HelpEs

La Vista simbólica varía de una aplicación a otra, pero su finalidad es la de almacenar definiciones simbólicas, tanto si son funciones, sentencias abiertas en x e y , o definiciones de análisis estadísticos. Utilice la Vista simbólica para definir funciones y sentencias abiertas, crear objetos geométricos, configurar una prueba de hipótesis y definir análisis estadísticos.

Observe que la Vista simbólica no se usa en las aplicaciones de Hoja de cálculo, Explorador, Solucionador de triángulo o Solucionador lineal.

0021 Config. simbólica

ID 21 · título localizado en HelpEs

La única vista común a todas las aplicaciones es la vista Config. simbólica. Su finalidad principal es la de permitirle anular tres de los ajustes del sistema especificados en la ventana Configuración de Inicio.

Pulse un ajuste una vez para mostrar un menú de opciones y, a continuación, seleccionar la opción que desea. También puede utilizar la rueda para resaltar un campo y tocar [[Selec.]] para mostrar el menú de opciones.

0022 Vista de gráfico

ID 22 · título localizado en HelpEs

La Vista de gráfico varía de una aplicación a otra pero su función es mostrarle las representaciones gráficas de los objetos definidos en la Vista simbólica.

En la Vista de gráfico puede ver los gráficos de las funciones y sentencias abiertas definidas en la Vista simbólica. En la aplicación Geometría puede crear objetos geométricos. Puede explorar funciones lineales, cuadráticas y sinusoidales, así como gráficos de amortización.

En la mayoría de las Vistas de gráfico, los gestos y las características siguientes están disponibles:

- toque para que el trazador pase a un valor x
- toque con el dedo para iniciar el desplazamiento cinético en la dirección deseada
- arrastre para desplazar la ventana
- abra/cierre los dedos en diagonal (coloque dos dedos en la pantalla y sepárelos o júntelos) para acercar o alejar
- gesto de pinza horizontal abierto/cerrado para aumentar o reducir el tamaño del eje x
- gesto de pinza vertical abierto/cerrado para aumentar o reducir el tamaño del eje y
- pulse + para acercar o - para alejar el zoom en la posición del cursor

Puede presionar Shift View (Copy) para copiar la Vista de gráfico actual en el portapapeles como una imagen.

Tenga en cuenta que la Vista de gráfico no se utiliza en las aplicaciones Hoja de cálculo, Solucionador de triángulo o Solucionador lineal.

0023 Config. de gráfico

ID 23 · título localizado en HelpEs

Esta vista se usa principalmente para modificar la apariencia de los gráficos y del entorno de gráficos. No se usa en aplicaciones que no tienen la Vista de gráfico. En las aplicaciones con esta vista, la Página 1 de la vista cuenta con campos para el rango de X, el rango de Y y el espacio de las marcas en los ejes.

La Página 2 se dedica a diversas opciones como las líneas de la cuadrícula, los puntos de la cuadrícula, los ejes, las etiquetas de los ejes y el tipo de cursor.

Algunas aplicaciones de HP Prime admiten el uso de una imagen como fondo en la Vista de gráfico. En esas aplicaciones, la tercera página de la Configuración de gráfico se dedica a la selección de la imagen y la configuración de su apariencia en la Vista de gráfico. Presione Shift-Plot para abrir la Configuración de gráfico y, a continuación, use la tecla del menú [-Página */*-] para avanzar hasta llegar a la página de Imagen como fondo.

El primer campo es una lista desplegable con las siguientes opciones:

- Sin fondo: no se usará ninguna imagen como fondo (opción predeterminada)
- Centrada: la imagen seleccionada se centrará, vertical y horizontalmente, en la Vista de gráfico
- Extendida: la imagen seleccionada se extenderá, tanto horizontal como verticalmente (si es necesario), para ocupar toda la pantalla en la Vista de gráfico
- Mejor ajuste: la imagen seleccionada se extenderá, horizontal o verticalmente (pero no en ambos sentidos), para ocupar una de las dos dimensiones en la Vista de gráfico
- Rango de XY: el usuario introducirá el rango de x y de y para posicionar la imagen en la Vista de gráfico

Al lado del primer campo está el campo Opacidad. Introduzca un entero de 0 a 100 para indicar el nivel de opacidad de la imagen, donde 0 es transparente y 100 es totalmente opaco.

Debajo de estos dos campos hay un selector con deslizamiento. El selector con deslizamiento muestra todas las imágenes asociadas con la primera aplicación actual, seguido de todas las imágenes incorporadas. Deslice para ver la biblioteca de imágenes disponibles, luego toque para seleccionar la que quiere. Una vez que haya seleccionado una opción de visualización, un nivel de opacidad y una imagen, la imagen seleccionada estará en la Vista de gráfico como fondo, con las opciones y el nivel de opacidad que eligió.

Si arrastra los ejes o hace el gesto de pinza para aumentar o reducir la Vista de gráfico solo se ve afectada la vista de la imagen si ha elegido la opción Rango de XY. Para todas las otras opciones, la imagen es un fondo verdadero y no responde a los cambios en la Vista de gráfico. En Rango XY, la imagen se mantiene en los rangos de x y de y a

medida que usa el zoom o el paneo, lo que le permite aumentar o reducir la imagen o recorrer una característica particular.

Hay dos teclas especiales del menú en la Página 3 de la Configuración Gráfico:

- [[Más]] toque para extraer una imagen que quiere de otra aplicación de HP Prime
- [[Calc]] con la opción Rango XY, úsela para calcular la cuarta coordenada, dadas las otras tres

Más

Toque la tecla de menú [[Más]] para ver un formulario de entrada con una lista desplegable y un selector con deslizamiento. Use la lista desplegable para seleccionar una aplicación de HP Prime que actualmente tenga imágenes asociadas con ella. Luego, use el selector con deslizamiento para recorrer las imágenes asociadas con una aplicación y toque para seleccionar una. Toque la tecla de menú [[OK]] para poner esa imagen en su aplicación actual.

Calc

Si ha elegido la opción RANGO XY, complete 3 de los 4 valores del rango x y del rango y, luego seleccione el cuarto y toque en la tecla de menú [[Calc]]. El valor del cuarto campo se calculará para usted, manteniendo la relación de aspecto de la imagen original intacta.

Para obtener información sobre cómo asociar una imagen con una aplicación de HP Prime, consulte la Guía de usuario del kit de conectividad de HP.

0024 Vista numérica

ID 24 · título localizado en HelpEs

La Vista numérica varía de una aplicación a otra, pero su finalidad es la presentar conjuntos de valores numéricos, ya sean valores de función o datos numéricos.

En la Vista numérica puede explorar tablas de valores generados por funciones, realizar mediciones geométricas, realizar cálculos de hojas de cálculo e introducir datos para análisis estadísticos.

En la mayoría de las Vistas numéricas, se admiten los siguientes gestos y funciones:

- toque para seleccionar un valor x o y. Puede copiar el valor en el portapapeles y luego pegarlo en cualquier lugar. Si selecciona un valor de la variable independiente, puede escribir un número real y la tabla se vuelve a configurar en base a ese valor.
- mantenga pulsado y luego arrastre para seleccionar una matriz rectangular de números. A continuación, puede copiar la matriz y pegarla en las aplicaciones de Hoja de cálculo, Var 1 estadística o Var 2 estadísticas, o en los Editores de lista o de matriz.
- toque con el dedo para iniciar el desplazamiento cinético en la dirección deseada
- arrastre para desplazar la ventana
- abra los dedos verticalmente para acercar la fila de la tabla actualmente seleccionada
- cierre los dedos verticalmente para alejar la fila de la tabla actualmente seleccionada
- presione + para acercar o - para alejar el zoom en la fila de la tabla seleccionada actualmente, utilizando el factor de zoom ajustado en la Configuración numérica

Tenga en cuenta que la Vista numérica no se utiliza en las aplicaciones Explorador.

0025 Vista Configuración numérica

ID 25 · título localizado en HelpEs

La Vista Configuración numérica se utiliza para determinar la apariencia de la Vista numérica y configurar el factor de zoom.

0026 Info

ID 26 · título localizado en HelpEs

Editor de nota de la aplicación : Shift + Apps (Info)

Use este editor para agregar una nota a la aplicación actual (o modificar la nota existente). La nota creada aquí se mantiene con esta aplicación cuando se transfiere a una PC u otra HP Prime. Este editor tiene la misma funcionalidad del Editor de notas.

Los elementos de menú son los siguientes:

- [[Editar]] toque para agregar una nota nueva o editar la nota real.

- [[Formato]] muestra un menú de opciones de formato
- [[Estilo]] muestra un menú de opciones de estilo
- [-▲ Página ▼-] le permite desplazarse de una página a otra en una nota de varias páginas
- [[●]] pasa por los estilos de las viñetas
- [[Insert]] toque para mostrar un menú de elementos que se pueden insertar.

Pulse dos veces ALPHA para bloquear el desplazamiento alfa. Pulse de nuevo para volver a activarlo.

Puede copiar y pegar texto usando Shift View (Copy) y Shift Menu (Paste) respectivamente.

0027 Tecla View

ID 27 · título recuperado de la página española

Presione la tecla View para abrir el menú Vistas. El menú Vistas contiene opciones específicas de las aplicaciones de gráfico, como diseños de pantalla dividida y opciones de escala utilizadas habitualmente.

- P. divid.: det. de gráf. Divide la pantalla en dos paneles, ambos con la Vista de gráfico. Las operaciones de zoom solo afectan al gráfico mostrado en el panel derecho.
- P. div.: tabla de gráf. Divide la pantalla en dos paneles, con la Vista de gráfico en uno y la Vista numérica en el otro.

Opciones de zoom habituales:

- Escala automática: ajusta el rango vertical (Rango Y) de forma que, al menos, una de las definiciones seleccionadas en la Vista simbólica se vea claramente
- Decimales: cambia la escala de ambos ejes de forma que cada píxel represente 0,1 unidades.
- Entero: cambia la escala del eje horizontal de forma que cada píxel represente 1 unidad.
- Trig: cambia la escala del eje horizontal de forma que cada píxel represente $\pi/24$ radianes, o 7,5 grados, y la del eje vertical de forma que cada píxel represente 0,1 unidades.

También puede añadir entradas personalizadas al menú Vistas mediante la definición de instrucciones VIEW en programas asociados a aplicaciones. Para obtener más información, consulte la referencia de VIEW.

1531 Imagen de fondo

ID 1531 · título localizado en HelpEs

Este cuadro de diálogo está disponible como la última página de la vista de configuración de gráfico de la mayoría de las aplicaciones que tienen una vista de gráfico.

El cuadro de selección de la parte superior le permite elegir la imagen que aparecerá en el fondo de la vista de gráfico de la aplicación. La lista contiene todas las imágenes incorporadas en la calculadora para esta aplicación, además de cualquier otra imagen que haya guardado en el archivo de la aplicación para esta aplicación (por ejemplo, mediante del kit de conectividad).

La casilla de verificación a la derecha de la selección de imagen le permite elegir si desea que la imagen sea un fondo en pantalla completa (sin marcar) o si desea especificar las coordenadas cartesianas donde se extraerá la imagen.

Puede controlar la opacidad de la imagen (100 significa una imagen sólida, cuanto más bajo el número, mayor será la disolución de la imagen en el fondo).

Si se selecciona una imagen almacenada en un archivo de la aplicación, puede eliminarla tocando el menú [[Eliminar]] (seleccione el nombre de la the imagen para que aparezca el menú)

Puede seleccionar cualquiera de las 4 coordenadas de la imagen y tocar el menú [[Calcular]] para recalcular esta coordenada de forma que se conserve la relación de aspecto de la imagen.

Puede importar imágenes de las otras aplicaciones tocando el menú [[Impor.]].

1532 Importación de imágenes

ID 1532 · título localizado en HelpEs

Este cuadro de diálogo le permite seleccionar cualquier imagen de cualquiera de las otras aplicaciones (incorporada o creada por el usuario) e importarla como archivo de usuario en la aplicación actual.

Seleccione la imagen que quiere importar usando el cuadro de selección.

Puede cambiar el nombre del archivo de destino para la imagen.

Puede cambiar el tamaño de la imagen.

Si la casilla de verificación de Proporción está marcada, el sistema mantendrá la relación de ancho/altura de la imagen al redimensionarla.

Cuando presione [[OK]], la imagen redimensionada se copiará en la aplicación actual usando el nombre especificado.

0028 Variables de aplicaciones comunes

ID 28 · título localizado en HelpEs

Las variables de aplicaciones almacenan la configuración y los resultados actuales en sus respectivas aplicaciones. Por ejemplo, la configuración Mrc X se almacena en una variable, al igual que la configuración de medida del ángulo y del método de creación de gráficas.

Puede cambiar una configuración almacenando un valor en la variable de una aplicación o comprobar una configuración evaluando la variable de una aplicación y comprobando su valor.

Las variables comunes a la mayoría de las aplicaciones se enumeran en esta sección. En el caso de variables específicas de una aplicación particular, consulte la sección dedicada a esa aplicación.

0029 Variables comunes de la Vista de gráfico

ID 29 · título localizado en HelpEs

Esta sección enumera las variables comunes a muchas de las aplicaciones de HP que tienen una Vista de gráfico.

Las variables de Gráfico incluyen, entre otras variables específicas de la aplicación, Xmin, Xmax, Xtick, Ymin, Ymax y Ytick.

No todas las aplicaciones tienen variables de la Vista de gráfico.

0030 Variable de aplicación Xmin

ID 30 · título recuperado de la página española

Xmin establece el valor mínimo horizontal de la Vista de gráfico.

Xmin := n, donde n es un número real

0031 Variable de aplicación Xmax

ID 31 · título recuperado de la página española

Xmax establece el valor máximo horizontal en la Vista de gráfico.

Xmax := n, donde n es un número real, de forma que $n > Xmin$

0032 Variable de aplicación Ymin

ID 32 · título recuperado de la página española

Ymin establece el valor mínimo vertical de la Vista de gráfico.

Ymin := n, donde n es un número real

0033 Variable de aplicación Ymax

ID 33 · título recuperado de la página española

Ymax establece el valor máximo vertical en la Vista de gráfico.

Ymax := n, donde n es un número real de forma que $n > Ymin$

0034 Variable de aplicación Xtick

ID 34 · título recuperado de la página española

Xtick establece la distancia entre las marcas de verificación del eje horizontal.

Xtick := n, donde n es un número real, de forma que $n > 0$

0035 Variable de aplicación Ytick

ID 35 · título recuperado de la página española

Ytick establece la distancia entre las marcas de verificación del eje vertical.

Ytick := n, donde n es un número real de forma que $n > 0$

0036 Variable de aplicación Axes

ID 36 · título recuperado de la página española

Activa o desactiva la visualización de los ejes X e Y en la Vista de gráfico.

Axes := 0 ejes activados (opción predefinida)

Axes := 1 ejes desactivados.

0037 Variable de aplicación Labels

ID 37 · título recuperado de la página española

Labels habilita o deshabilita las etiquetas de dibujo en la Vista de gráfico que muestra los rags de X y de Y.

Labels := 0 sin etiquetas (opción predeterminada)

Labels := 1 con etiquetas

0038 Variable de la aplicación GridDots

ID 38 · título recuperado de la página española

GridDots activa o desactiva los puntos de la cuadrícula del fondo en la Vista de gráfico.

GridDots := 0 puntos de cuadrícula activados (opción predeterminada)

GridDots := 1 puntos de cuadrícula desactivados.

0039 Variable de la aplicación GridLines

ID 39 · título recuperado de la página española

GridLines activa o desactiva las línea de cuadrícula del fondo en la Vista de gráfico.

GridLines := 0 líneas de cuadrícula activadas (opción predeterminada)

GridLines := 1 líneas de cuadrícula desactivadas.

0040 Variable de aplicación cursor

ID 40 · título recuperado de la página española

Cursor establece el tipo de cursor en la Vista de gráfico.

Cursor := 0 normal (opción predeterminada)

Cursor := 1 invertido

Cursor := 2 intermitente

0041 Variable de la aplicación ImageName

ID 41 · título recuperado de la página española

ImageName controla cuál imagen se establece como fondo en las vistas de gráfico.

ImageName := nombre, donde el nombre es una cadena de nombre de archivo (por ejemplo "foto1").

0042 Variable de la aplicación ImageDisplay

ID 42 · título recuperado de la página española

ImageDisplay controla cómo aparece una imagen de fondo

0 sin fondo

1 centrada

2 extendida

3 el mejor ajuste

4 rango XY

ImageName controla cuál imagen se usa

0043 Variable de la aplicación ImageOpacity

ID 43 · título recuperado de la página española

ImageOpacity controla la opacidad de la imagen de fondo, si la hay. Use 100 en el caso de una imagen no alterada y valores más bajos para obtener menos opacidad (más transparencia) en la imagen.

ImageOpacity := n, donde n es un número real entre 0 y 100.

0044 Variable de la aplicación ImageXmin

ID 44 · título recuperado de la página española

ImageXmin establece el valor horizontal mínimo ocupado por la imagen de fondo, si la hay.

ImageXmin := n, donde n es un número real.

0045 Variable de la aplicación ImageXmax

ID 45 · título recuperado de la página española

ImageXmax establece el valor horizontal máximo ocupado por la imagen de fondo, si la hay.

ImageXmax := n, donde n es un número real.

0046 Variable de la aplicación ImageYmin

ID 46 · título recuperado de la página española

ImageYmin establece el valor vertical mínimo ocupado por la imagen de fondo, si hay una.

ImageYmin := n, donde n es un número real.

0047 Variable de la aplicación ImageYmax

ID 47 · título recuperado de la página española

ImageYmax establece el valor vertical máximo ocupado por la imagen de fondo, si hay una.

ImageYmax := n, donde n es un número real.

0048 Variable de la aplicación PlotMethod

ID 48 · título recuperado de la página española

PlotMethod establece el método de elaboración de gráficos:

PlotMethod := 0, Adaptable (opción predeterminada): brinda los resultados más precisos pero tarda más en producir el gráfico

PlotMethod := 1, Segmentos de incremento fijo: este método muestrea los valores de x, calcula sus valores de y correspondientes y luego traza y conecta los puntos.

PlotMethod := 2, Puntos de incremento fijo: este funciona como el método de segmentos de incremento fijo pero no conecta los puntos.

0049 Variable de aplicación Recenter

ID 49 · título recuperado de la página española

Recenter especifica si el gráfico se vuelve a centrar en el cursor durante las operaciones de Zoom en la Vista de gráfico.

Recenter := 0 — se vuelve a centrar en el cursor (opción predeterminada)

Recenter := 1 — no se vuelve a centrar en el cursor

0050 Variable de aplicación Xzoom

ID 50 · título recuperado de la página española

Xzoom establece el factor de zoom horizontal.

Xzoom := n, donde n es un número real de forma que $n > 0$ (el valor predeterminado es 2)

0051 Variable de aplicación Yzoom

ID 51 · título recuperado de la página española

Yzoom establece el factor de zoom vertical.

Yzoom := n, donde n es un número real de forma que $n > 0$ (el valor predeterminado es 2)

0052 Variable de aplicación Tmin

ID 52 · título recuperado de la página española

Tmin contiene el valor inicial de T en la vista de gráfico de Paramétrica.

Tmin := n, donde n es un número real

0053 Variable de aplicación Tmax

ID 53 · título recuperado de la página española

Tmax contiene el valor final de T en la Vista de gráfico de Paramétrica.

Tmax := n, donde n es un número real de forma que $n > T_{min}$

0054 Variable de aplicación Tstep

ID 54 · título recuperado de la página española

Tstep contiene el valor de paso (incremento) de T en la vista de gráfico paramétrica.

Tstep := n, donde n es un número real de forma que $n > 0$

0055 Variable de aplicación Nmin

ID 55 · título recuperado de la página española

Nmin contiene el valor inicial para N en la vista Gráfico de la aplicación Secuencia.

Nmin := n, donde n es un número natural.

0056 Variable de aplicación Nmax

ID 56 · título recuperado de la página española

Nmax contiene el valor final de N en la vista Plot de la app Sequence.

Nmax := n, donde n es un número natural tal que $n > N_{min}$.

0057 Variable de aplicación θmin

ID 57 · título recuperado de la página española

θmin contiene el valor inicial para θ en la Vista de gráfico de la aplicación Polar.

θmin := n, donde n es un número real

0058 Variable de aplicación θmax

ID 58 · título recuperado de la página española

θmax contiene el valor final para θ en la Vista de gráfico de la aplicación Polar.

θmax := n, donde n es un número real de modo que $n > \theta_{min}$

0059 Variable de la aplicación θstep

ID 59 · título recuperado de la página española

θstep contiene el valor escalonado (incremento) de θ en la Vista de gráfico de la aplicación Polar.

θstep := n, donde n es un número real de modo que $n > 0$

0060 Variables comunes de la Vista numérica

ID 60 · título localizado en HelpEs

Esta sección enumera las variables comunes a muchas de las aplicaciones de HP que tienen una Vista numérica.

0061 Variable de aplicación NumStart

ID 61 · título recuperado de la página española

NumStart establece el valor inicial de la variable independiente en la Vista numérica cuando Automático es el Tipo de número.

NumStart := n, donde n es un número real

0062 Variable de aplicación NumStep

ID 62 · título recuperado de la página española

NumStep establece el tamaño del paso (incremento) de la variable independiente en la Vista numérica cuando Automático es el Tipo de número.

NumStep := n, donde n es un número real de forma que $n > 0$

0063 Variable de aplicación NumType

ID 63 · título recuperado de la página española

Determina cómo se generan los valores de la variable independiente en la Vista numérica:

- Automático: usa los valores de NumStart y NumStep para crear los valores de la variable independiente.
- Generar propio: permite introducir los valores de la variable independiente uno a uno.

NumType := 0 para automático (por defecto)

NumType := 1 para BuildYourOwn

0064 Variable de aplicación NumIndep

ID 64 · título recuperado de la página española

NumIndep contiene la lista de los valores que ha introducido en la Vista numérica cuando ha elegido el modo BuildYourOwn. En el caso de la aplicación Gráficos avanzados, la lista incluye los pares de valores.

0065 Variable de aplicación NumZoom

ID 65 · título recuperado de la página española

NumZoom establece el factor de la Vista numérica.

NumZoom := n, donde $n > 0$ (el valor predeterminado es 4)

0066 Lista de aplicaciones

ID 66 · título recuperado de la página española

Sintaxis

Apps ()

Devuelve una lista de las aplicaciones instaladas, incluidas las integradas. El primer elemento de esta lista es siempre la aplicación activa actual.

0067 Eliminar aplicación

ID 67 · título recuperado de la página española

Sintaxis

DeLApp (AppName)

Acepta el nombre de una aplicación como una cadena de texto. Para una aplicación definida por el usuario, la aplicación se elimina y se devuelve 0. Para una aplicación integrada, la aplicación se restablece y se devuelve 1. Para una aplicación inexistente, se devuelve 'Error: entrada no válida'.

0068 Archivos de aplicación

ID 68 · título recuperado de la página española

Sintaxis

```
AFiles()  
AFiles(filename)  
AFiles(filename):=object
```

Cada aplicación de HP Prime puede tener cualquier número de archivos asociados. Estos archivos se agrupan y se envían con la aplicación, lo que permite la persistencia y gestión de datos dentro del entorno de la aplicación.

AFiles guarda y lee objetos, lo que significa cualquier cosa que se pueda almacenar en una variable. Para acceder a un archivo en modo binario, use AFilesB.

Ejemplos

[Ejemplo gráfico E1] guarda la lista en el archivo 'data' asociado con una aplicación. Si el archivo no existe, se creará.
[Ejemplo gráfico E2] devuelve el contenido del archivo 'data' de una aplicación.
[Ejemplo gráfico E3] devuelve una lista de todos los nombres de archivos asociados con una aplicación.
[Ejemplo gráfico E4] devuelve una lista de todos los nombres de archivos de la aplicación 'MyApp'.

0069 Archivos de aplicación binaria

ID 69 · título recuperado de la página española

Sintaxis

```
AFilesB()  
AFilesB(Filename [, Pos [, Byte_count]])  
AFilesB(Filename [, Pos]):=Bytes
```

Cada aplicación de HP Prime puede tener cualquier número de archivos asociados. Estos archivos se agrupan y se envían con la aplicación, lo que permite la persistencia y gestión de datos dentro del entorno de la aplicación.

AFilesB sirve como el equivalente binario de AFiles, permitiendo operaciones de lectura y escritura a nivel de byte en archivos.

Parámetros

- Filename: El nombre de un solo archivo o una lista de nombres de archivos a los que se aplica la operación.
- Pos: La posición del primer byte en el archivo desde la cual comienza la operación. Comienza desde 0 (el primer byte del archivo), que también es el valor predeterminado.
- Byte_count: El número de bytes a leer, el valor predeterminado es 1.
- Bytes: Una lista de bytes para escribir en el archivo, donde cada byte se representa como un valor numérico que va de 0 a 255.

Ejemplos

[Ejemplo gráfico E1] devuelve la lista de todos los nombres de archivos asociados con una aplicación.
[Ejemplo gráfico E2] devuelve el tamaño en bytes del archivo data.
[Ejemplo gráfico E3] devuelve una lista de tamaños de todos los archivos asociados con una aplicación.
[Ejemplo gráfico E4] devuelve una lista de 2 bytes leídos del archivo data, comenzando desde la posición 4.
[Ejemplo gráfico E5] guarda 3 bytes, comenzando en la posición 2, en el archivo data.
[Ejemplo gráfico E6] devuelve una lista de todos los nombres de archivos de la aplicación MyApp.

0070 Eliminar archivos de la aplicación

ID 70 · título recuperado de la página española

Sintaxis

```
DelAFiles("nombre")  
DelAFiles(lista)
```

Elimina el archivo especificado asociado con una aplicación cuando se proporciona un nombre de archivo. Si se proporciona una lista de nombres de archivos, se eliminarán todos los archivos especificados.

0071 Nota de la aplicación

ID 71 · título recuperado de la página española

A`Note` devuelve la nota asociada con una aplicación HP. Esta es la nota mostrada cuando el usuario presiona Shift Apps (Info).

A`Note`:="string" establece la nota asociada con la aplicación para contener la cadena.

0072 Programa de la aplicación

ID 72 · título recuperado de la página española

A`Program` devuelve el programa asociado con una aplicación de HP Prime.

A`Program`:="string" establece el programa asociado con la aplicación para contener la cadena.

0073 Variables de aplicación

ID 73 · título recuperado de la página española

A`Vars` devuelve la lista de los nombres de todas las variables asociadas con una aplicación de HP Prime.

A`Vars`(n) devuelve el contenido de la *n*-ésima variable asociada con la aplicación.

A`Vars`("name") devuelve el contenido de la variable especificada asociada con la aplicación.

A`Vars`(n o "name"):= value establece el contenido de la variable de aplicación especificada en el valor dado. Si "name" no es una variable existente, crea una nueva.

Tenga en cuenta que, una vez creada una variable de aplicación mediante A`Vars`("name"):= value, puede usarla simplemente escribiendo el nombre de la variable.

0074 Eliminar las variables de la aplicación

ID 74 · título recuperado de la página española

Sintaxis

De`LAVars`(n)

De`LAVars`("name")

Borra la variable de aplicación especificada.

0075 Variables comunes del modo aplicaciones

ID 75 · título localizado en HelpEs

Esta sección enumera las variables de modo utilizadas por las aplicaciones de HP. Se encuentran en la Vista de configuración simbólica de cada aplicación.

Si el valor se establece en 0, se utilizará la configuración de la vista de Inicio. De lo contrario, el ajuste de la Vista de configuración simbólica de la aplicación reemplaza la configuración de la Vista de Inicio.

0076 Variable de la aplicación AAngle

ID 76 · título recuperado de la página española

A`Angle` establece el modo de ángulo en Grados, Radianes, o Sistema, para la aplicación.

A`Angle` := 0 usa la configuración de la Vista de Inicio

A`Angle` := 1 para Radianes

A`Angle` := 2 para Grados

A`Angle` := 3 for Gradianes

0077 Variable de la aplicación ADigits

ID 77 · título recuperado de la página española

A`Digits` establece la cantidad de lugares decimales que se van a usar en el formato del número.

A`Digits` := n, donde n es un entero de forma que $0 \leq n \leq 11$

0078 Variable de aplicación AFormat

AFormat define el formato de presentación del número para la aplicación.

AFormat := 0 usa la configuración de la Vista de Inicio

AFormat := 1 estándar

AFormat := 2 fijo

AFormat := 3 científico

AFormat := 4 de ingeniería

AFormat := 5 flotante

AFormat := 6 redondeado

0079 Variable de aplicación AComplex

AComplex establece el modo de número complejo para la aplicación.

AComplex := 0 usa la configuración de Vista de Inicio

AComplex := 1 activado

AComplex := 2 desactivado

0080 Aplicación Función

La aplicación Función permite explorar hasta 10 funciones de valores reales de la fórmula $Y=f(X)$ (por ejemplo, $y=2x+3$).

Una vez definida una función, puede:

- Ver su representación gráfica en el plano cartesiano
- Buscar raíces, puntos de intersección, pendiente, área firmada y extremos
- Crear tablas de valores de funciones
- Acercar o alejar el zoom en la representación

Para iniciar la aplicación Función, vaya a la biblioteca de aplicaciones y pulse el icono de la aplicación Función. También puede utilizar la rueda para seleccionar el icono de la aplicación Función, luego pulse [[Inicio]] o presione Enter para iniciar la aplicación.

0081 Vista simbólica Función

En la Vista simbólica de Función, es posible definir hasta diez funciones: $F1(X)$ a $F9(X)$ y $F0(X)$. Resalte uno de los campos de función y comience a introducir una expresión dependiente de x , o bien toque [[Editar]] para editar una expresión existente.

Cada expresión tiene un selector de color a la izquierda de su campo. Toque el cuadro de color para elegir el color de la gráfica correspondiente a dicha expresión.

Los botones del menú son:

- [[Editar]] abre una línea de entrada para editar la definición de la función seleccionada
- [[✓]] activa o desactiva la función seleccionada para su representación gráfica y la creación de tablas
- [[X]] ayuda de escritura para introducir funciones de x
- [[Mostr.]] muestra la función seleccionada en modo de pantalla completa con el desplazamiento horizontal y vertical habilitado.
- [[Eval]] resuelve las referencias a otras funciones, como $F2(X)=F1(X-1)+2$

0082 Vista de gráfico de Función

La Vista de gráfico sirve para ver y examinar las gráficas de las funciones definidas en la Vista simbólica. Toque [[Menú]] para activar y desactivar el menú

Los botones del menú son:

- **[[Zoom]]** abre el menú Zoom, con opciones para acercar, alejar, etc.
- **[[Trazar]]** permite activar o desactivar el curso de trazado.
- **[[Ir a]]** le permite mover el cursor a cualquier punto de la curva introduciendo su coordenada x.
- **[[Hacer b]]** te permite esbozar una función con tu dedo, y HP Prime la coincidirá con un gráfico de la función (detalles abajo).
- **[[Func.]]** permite abrir el menú Función con los siguientes comandos: Hacer boceto...: haga un boceto de una función con su dedo y HP Prime lo hará coincidir con un gráfico de la función Definición: muestra la definición de la función que se está trazando, con las opciones de editar la expresión o transformar el gráfico Transformación: arrastre para trasladar o haga el gesto de pinza para escalar la función actual. Raíz: encuentra la raíz de la función actual más próxima del trazador Intersección: encuentre la intersección de la función actual con otra función, más cercana al trazador Pendiente: encuentre la pendiente de la función actual en la ubicación actual del trazador Área firmada...: encuentre el área con signo debajo de una curva o entre dos de ellas Extremo: encuentre un extremo para la función actual, más próximo a la ubicación del trazador Tangente: dibuja la tangente de la función actual mediante el punto de trazado actual
- **[[Menú]]** activa o desactiva el menú

Hacer boceto, Transformación y Definición

Seleccione Boceto para poner la Vista de gráfico en modo Boceto. Haga un boceto de una función simple (lineal, cuadrada, etc.) con su dedo. Cuando levante su dedo de la pantalla, su boceto se sustituirá por el gráfico más cercano que HP Prime puede administrar; toque **[[OK]]** y su expresión se agregará a la Vista simbólica.

Seleccione Transformación para trasladar y extender el gráfico de la función actual. Arrastre para trasladar o hacer gesto de pinza vertical u horizontalmente, con el fin de ampliar el gráfico. La expresión responderá como corresponda. Pulse **[[Simplificar]]** para simplificar la expresión actual. Pulse **[[Forma]]** para seleccionar una forma alternativa para la ecuación de su función.

Seleccione Definición para ver la expresión del gráfico actual. En el cuadro de Definición, puede pulsar **[[Editar]]** para editar la expresión. Cuando haya terminado de editar, pulse **[[OK]]** para ver el cambio en el gráfico. También puede pulsar **[[Transf.]]** para manipular el gráfico actual. Pulse **[[OK]]** cuando haya terminado para volver al cuadro de Definición. Vuelva a pulsar el menú desplegable para cerrar el cuadro.

Utilice las teclas izquierda y derecha del cursor o toque para realizar el trazado de una función. Utilice las teclas arriba y abajo del cursor para cambiar de una función a otra. Pulse **+** para acercar el zoom en la ubicación actual del cursor y pulse **-** para alejar el zoom. Configure el factor de zoom en el menú Zoom.

También puede utilizar todos los gestos comunes a la Vista de gráfico. Consulte Vista de gráfico para obtener más detalles.

0083 Configuración de gráfico de Función

ID 83 · título localizado en HelpEs

Config. de gráfico de Función permite controlar la apariencia de la ventana de la gráfica, incluida la apariencia del cursor, si deben dibujarse o no los ejes, etc. Esta configuración tiene dos páginas.

En la primera página, los campos son:

- Rng X: el rango de creación de gráfico horizontal
- Rng Y: el rango de creación de gráfico vertical
- Mrc X: espaciado entre marcas de graduación horizontal
- Mrc Y: espaciado entre marcas de graduación vertical

Los botones del menú en la primera página son:

- **[[Editar]]** abre una línea de edición para editar el valor del campo seleccionado
- **[-Página 1/2 ▼-]** muestra la segunda página de la configuración

En la segunda página, los campos son:

- Ejes: activa o desactiva los ejes
- Etiquetas: activa o desactiva las etiquetas del eje
- Puntos cuadr.: activa o desactiva los puntos de la cuadrícula
- Líneas cuadríc.: activa o desactiva las líneas de la cuadrícula
- Cursor: permite elegir entre cursores estándar, invertido y parpadeante
- Método: permite elegir entre adaptable, Segmento de increm. fijo y Puntos de increm. fijo

Los botones del menú en la segunda página son:

- [[✓]] activa o desactiva la configuración actual
- [[Selec.]] permite elegir de un cuadro de elección
- [-▲ Página 2/2-] retorna a la primera página de la configuración

El campo Método requiere una explicación. De forma predeterminada, la HP Prime utiliza el método Adaptable, un método avanzado que da resultados muy precisos. Puede elegir el método más tradicional, denominado Segm. de increm. fijo, que muestra los valores x, calcula sus valores y correspondientes, y a continuación, traza los puntos y los conecta. O bien, puede seleccionar Puntos de increm. fijo, que funciona como Segm. de increm. fijo pero no conecta los puntos.

0084 Vista numérica de Función

ID 84 · título localizado en HelpEs

La Vista numérica de Función se ha diseñado para crear y examinar tablas de valores X/Y, con base en las funciones definidas en la Vista simbólica.

Toque en cualquier fila de la columna x, introduzca un valor real y toque [[OK]]. La configuración de la tabla cambiará. También puede acercar o alejar el zoom en cualquier fila de la tabla. Pulse + para acercar el zoom en una fila de la tabla y - para alejar el zoom.

Los botones del menú son:

- [[Zoom]] permite acercar o alejar el zoom en una fila de la tabla. Tenga en cuenta que en la Vista numérica, el zoom cambia el incremento entre los valores x consecutivos. Acercar el zoom disminuye el incremento; alejar el zoom aumenta el incremento. Los valores de la fila que acerque o aleje siguen siendo los mismos.
- [[Más]] abre un menú con opciones para seleccionar varias celdas; luego puede copiarlas y pegarlas en otro lugar.
- [[Ir a]] pasa a un valor determinado de la variable independiente
- [[Defn]] muestra la definición de la columna seleccionada

El menú [[Más]] incluye las siguientes opciones:

- Seleccionar Fila: selecciona la fila que contiene la celda seleccionada en ese momento; la fila se puede copiar y pegar en otros lugares Intercambia final: esta opción está disponible una vez hecha la selección de varias celdas. Intercambia las celdas del principio y el final de la selección actual. Incluir encabezados: lo mismo que Seleccionar fila, excepto que también se seleccionan los encabezados de la fila
- Selección: activa o desactiva el modo de selección
- Tamaño de la fuente: selecciona una fuente pequeña, mediana o de gran tamaño

También puede utilizar cualquiera de los gestos comunes a la Vista numérica. Consulte Vista numérica para obtener más detalles.

0085 Configuración numérica de Función

ID 85 · título localizado en HelpEs

Configuración numérica de Función permite controlar la apariencia de la tabla en la Vista numérica, incluido qué valor de X se encuentra al principio de la tabla, cuál es el intervalo entre valores de X y cuál es el factor de zoom utilizado para acercar y alejar el zoom en una fila de la tabla.

Los campos son los siguientes:

- Núm Inicial: el primer valor de x en la tabla
- Núm. increm.: el incremento entre valores x consecutivos
- Núm. Zoom: el factor de zoom para acercar o alejar
- Tipo de núm.: elegir entre tipos de tablas Automático: proporciona los valores de x- y de función- Generar propio: usted proporciona los valores x; la aplicación proporciona los valores de la función correspondientes

Los botones del menú son:

- [[Editar]] abre una línea de edición para editar el valor actual de un campo
- [[Selec.]] permite elegir de un cuadro de elección
- [[Gráf→]] permite configurar Núm. inicial y Núm. increm. de modo que los valores de variable independiente de la tabla de la Vista numérica coincidan con los valores de la variable independiente al realizar trazados en la Vista de gráfico.

0086 Variables de la aplicación Función

ID 86 · título localizado en HelpEs

Para mostrar las variables relacionadas con la aplicación Función, pulse Vars, toque [[Apl.]] y seleccione Función.

Las variables de la aplicación Función se agrupan en 5 categorías:

- Resultados
- Simbólica
- Gráfico
- Numerica
- Modos

Las variables de aplicación Gráfico, Numérico y Modos se analizan bajo Variables de aplicaciones comunes. La aplicación de la función simbólica Resultados y Simbólico se analizan en las siguientes secciones.

0087 Variables simbólicas

ID 87 · título localizado en HelpEs

Las variables simbólicas de la aplicación Función son F1 a F9 y F0. Estas variables contienen expresiones algebraicas en X.

$F_n := f(X)$, donde n es un entero entre 0 y 9 inclusive y $f(x)$ es una expresión algebraica en X.

0088 Variables de resultados

ID 88 · título localizado en HelpEs

Las variables de resultados de la aplicación Función almacenan resultados de los cálculos realizados en el menú del Fcn en la Vista de gráfico.

0089 Variable de la aplicación Extremum

ID 89 · título recuperado de la página española

Extremum contiene el último valor encontrado por la función Extremum en el menú Func. de Vista de gráfico.

0090 Variable de la aplicación Isect

ID 90 · título recuperado de la página española

Isect contiene el último valor encontrado por la función Intersección en el menú Func. de la Vista de gráfico.

0091 Variable de aplicación Root

ID 91 · título recuperado de la página española

Root contiene el último valor encontrado por la función Raíz en el menú Func. de la Vista de gráfico.

0092 Variable de aplicación SignedArea

ID 92 · título recuperado de la página española

SignedArea contiene el último valor encontrado en la función Área firmada en el menú Func. en la Vista de gráfico.

0093 Variable de aplicación Slope

ID 93 · título recuperado de la página española

Slope contiene el último valor encontrado por la función Slope en el menú Func. de la Vista de gráfico.

0094 Funciones de la aplicación Función

ID 94 · título localizado en HelpEs

Las funciones de la aplicación Función ofrecen la misma funcionalidad que la que proporciona la Vista de gráfico de dicha aplicación, en el menú Func. Todas estas operaciones se utilizan con funciones. Las funciones pueden ser expresiones de X o los nombres de las variables de la aplicación Función F0 a F9.

0095 Función de aplicación AREA

ID 95 · título recuperado de la página española

Sintaxis

AREA(Fn [, Fm], Inferior, Superior)

El área señalizada debajo de una curva o entre curvas. Aproxima el área señalizada bajo una función o entre dos funciones. Encuentra el área bajo la función Fn o debajo de Fn y sobre la función Fm, desde el Inferior valor de X hasta el Superior valor de X.

0096 Función de aplicación EXTREMUM

ID 96 · título recuperado de la página española

Sintaxis

EXTREMUM(Fn [, Supuesto])

Extremo de una función. Busca el extremo (si existe) de la función Fn más cercano al valor Supuesto de X.

0097 Función de aplicación ISECT

ID 97 · título recuperado de la página española

Sintaxis

ISECT(Fn, Fm [, Supuesto])

Intersección de dos funciones. Encuentra la intersección (si existe) de las dos funciones Fn y Fm que es la más cercana al valor de X Supuesto.

0098 Función de aplicación ROOT

ID 98 · título recuperado de la página española

Sintaxis

ROOT(Fn [, Supuesto])

Raíz de una función. Busca la raíz de la función Fn (si existe) más cercana al valor Supuesto de X.

0099 Función de aplicación SLOPE

ID 99 · título recuperado de la página española

Sintaxis

SLOPE(Fn, Valor)

Pendiente de una función. Devuelve la pendiente de la función Fn al valor de X Valor (si existe).

0100 Aplicación Gráficos avanzados

ID 100 · título localizado en HelpEs

La aplicación Gráficos avanzados permite definir y explorar los gráficos de sentencias simbólicas abiertas que dependen de x o y, de ambos o de ninguno. Puede trazar secciones cónicas, polinomios en formato estándar o general, desigualdades y funciones.

Cuando haya definido una sentencia abierta, puede trazarla y ver una tabla de valores que se muestra cuando se ha satisfecho la sentencia abierta.

Para iniciar la aplicación Gráficos avanzados, vaya a la Biblioteca de aplicaciones y pulse el icono de la aplicación Gráficos avanzados. También puede utilizar la rueda para seleccionar el icono de la aplicación Gráficos avanzados, luego pulse [[Iniciar]] o presione Enter para iniciar la aplicación.

GALERÍA DE GRÁFICO

Para obtener una idea de los tipos de gráficos que puede trazar con la aplicación Gráficos avanzados, abra la aplicación, vaya a la Vista de gráficos, presione la tecla Menú y elija Visitar galería de gráficos en el menú. Se abre una galería de gráficos, con las definiciones que se generaron se muestra un gráfico en la parte inferior de la pantalla. Presione la rueda a la izquierda/derecha para pasar de un gráfico a otro en la galería.

Para guardar un gráfico y explorarlo personalmente, pulse [[Guard.]] e introduzca un nombre para una nueva aplicación personalizada. A continuación puede explorar la nueva aplicación de la misma forma que exploraría la aplicación Gráficos avanzados integrada.

0101 Vista simbólica de Gráficos avanzados

ID 101 · título localizado en HelpEs

La Vista simbólica Gráficos avanzados permite definir hasta diez enunciados abiertos. Resalte uno de los diez campos de definición, etiquetados de V0 a V9, y comience a introducir o editar un enunciado abierto.

Cada definición tiene un selector de color a la izquierda de su campo. Toque el cuadro de color para elegir el color de la gráfica correspondiente a esa definición.

Botones de menú: VISTA PRINCIPAL

- [[Editar]] copia la definición seleccionada en la línea de entrada y activa el cursor
- [[✓]] selecciona o anula la selección de la definición resaltada. Solo se representan gráficamente las definiciones seleccionadas.
- [[X]] es una ayuda de escritura para introducir una X
- [[Y]] es una ayuda de escritura para introducir una Y
- [[Mostr.]] muestra la definición seleccionada en modo de pantalla completa con el desplazamiento horizontal y vertical habilitado
- [[Eval.]] resuelve las referencias cuando una definición se expresa en función de otra

Botones de menú: VISTA DE EDICIÓN

- [[=]] inserta el signo igual
- [[≤, ≥, ≠]] abre una paleta de operadores relacionales
- [[X]] es una ayuda de escritura para introducir una X
- [[Y]] es una ayuda de escritura para introducir una Y
- [[Canc.]] cierra la línea de edición sin guardar los cambios
- [[OK]] guarda los cambios y cierra la línea de edición

0102 Vista de gráfico de Gráficos avanzados

ID 102 · título localizado en HelpEs

La Vista de gráfico sirve para ver y examinar los gráficos de las sentencias abiertas definidas en la Vista simbólica. Tenga en cuenta que solo se trazan las definiciones seleccionadas (activadas) en la Vista simbólica.

Toque [[Menú]] para mostrar u ocultar otros botones de menú.

Botones de menú:

- [[Zoom]] permite abrir el menú Zoom, con opciones para acercar o alejar el zoom, etc.
- [[Trazar]] proporciona varias opciones de trazado: Desactivado Interior Extremo Pol. Selección
- [[Ir a]] permite mover el cursor a la ubicación especifique
- [[Defn]] muestra la definición de la sentencia abierta seleccionada (que luego puede editar sin la necesidad de volver a la Vista simbólica)
- [[Menú]] muestra u oculta los otros botones de menú

Se puede personalizar el trazador para recorrer el borde del gráfico, o para trazar los puntos de interés como las intercepciones de x, las intercepciones de y, extrema, o los puntos de inflexión.

Los gestos comunes a la Vista de gráfico son compatibles aquí también. Consulte Vista de gráfico para obtener más detalles.

0103 Trazado de gráficos en Gráficos avanzados

ID 103 · título localizado en HelpEs

La opción de trazado en la Vista de gráficos en la aplicación de Gráficos avanzados puede ajustarse para trazar cualquiera de las siguientes configuraciones:

- Valores de los extremos

- Puntos de interés, tales como: intersecciones de X intersecciones de Y Intersecciones con... (visible cuando se seleccionan más de un elemento en la vista simbólica) Extremo horizontal Extremo vertical Puntos de inflexión

En la Vista de gráfico, pulse la tecla de menú **[[Trazar]]** y realice una selección. Por ejemplo, si está trazando en la gráfica de $Y=\sin(X)$ y elige trazar intersecciones X, el trazador saltará de una intersección X a la siguiente. Si el trazador está en el punto (0, 0) y presiona la rueda hacia la derecha, el trazador se moverá a $(\pi, 0)$.

Las mismas opciones de trazado están disponibles en la Vista numérica, para que pueda configurar la tabla para mostrar los mismos valores que trazó en la Vista de gráfico.

0104 Defn de Gráficos avanzados en Vista de gráfico

ID 104 · título localizado en HelpEs

Puede editar la sentencia abierta que se muestra cuando toca **[[Defn]]** (es decir, puede editarla sin tener que ir a la Vista simbólica). Toque **[[Editar]]** y el cursor aparece al final de la definición. Realice los cambios y toque **[[OK]]** (o presione Enter). La gráfica cambiará para reflejar la nueva definición. (La definición original en la Vista simbólica también cambiará).

Para cerrar el panel de definición, pulse el botón de menú **[[↵]]**.

0105 Galería de gráfico para Gráficos avanzados

ID 105 · título localizado en HelpEs

La Galería de gráfico es una galería de gráficos que puede guardar como una nueva instancia de la aplicación Gráficos avanzados. Mientras se encuentra en la Vista de gráfico, presione la tecla Menú y seleccione Visitar la Galería de gráficos.

Presione la rueda hacia la izquierda/derecha para moverse de gráfico en gráfico. Tenga en cuenta que las definiciones que generó el gráfico actual se muestran en la parte inferior de la pantalla.

Para guardar un gráfico y explorarlo personalmente, pulse **[[Guard.]]** e introduzca un nombre para una nueva aplicación personalizada. A continuación puede explorar la nueva aplicación de la misma forma que exploraría la aplicación Gráficos avanzados integrada.

0106 Configuración de Vista de gráfico

ID 106 · título localizado en HelpEs

La Vista Config. de gráfico de la aplicación Gráficos avanzados permite controlar la apariencia de la Vista de gráfico, incluida la apariencia del cursor, si los ejes se dibujan, etc.. Esta vista tiene dos páginas.

Página 1

- Rng X: el rango de creación de gráfico horizontal
- Rng Y: el rango de creación de gráfico vertical
- Mrc X: espaciado entre marcas de graduación horizontal.
- Mrc Y: espaciado entre marcas de graduación vertical.

Página 2

- Ejes: activa o desactiva los ejes
- Etiquetas: activa o desactiva las etiquetas del eje
- Puntos cuadr.: activa o desactiva los puntos de la cuadrícula
- Líneas cuadríc.: activa o desactiva las líneas de la cuadrícula
- Cursor: permite elegir entre cursores estándar, invertido y parpadeante

0107 Vista numérica de Gráficos avanzados

ID 107 · título localizado en HelpEs

A diferencia de otras muchas aplicaciones de HP, la Vista numérica de la aplicación Gráficos avanzados no ofrece una tabla de valores de la variable dependiente como la generan las definiciones seleccionadas en la Vista simbólica. En lugar de eso, tanto x como y se ven como independientes y la Vista numérica muestra si cada sentencia abierta seleccionada se satisface o no para un conjunto de pares ordenados (x, y). Si se satisface, aparece Verdadero junto a la combinación; de lo contrario, aparece Falso.

Puede introducir sus propios valores en las columnas X e Y. La aplicación volverá a evaluar automáticamente si cada sentencia abierta que se representa en la Vista numérica satisface dichos valores.

Puede personalizar la tabla con la tecla de menú **[[Trazar]]**. En lugar de mostrar los valores x e y, y ya sea que satisfagan o no a la sentencia, puede optar por completar la tabla con las coordenadas de los puntos de interés que aparecen en la Vista de gráfico. Por ejemplo, puede seleccionar crear una tabla de la actual serie de interceptaciones de x, puntos de inflexión, o puntos de intersección visibles en la Vista de gráfico.

Botones de menú:

- **[[Zoom]]** permite abrir el menú Zoom, con opciones para acercar o alejar el zoom en la fila seleccionada de la tabla
- **[[Más]]** abre un menú con las opciones de edición
- **[[Trazar]]** proporciona varias opciones para lo que se desea mostrar en la tabla: Valores de los extremos Puntos de interés (POI)
- **[[Defn]]** muestra la definición de la sentencia abierta seleccionada

El menú **[[Más]]** incluye las siguientes opciones:

- **Seleccionar Fila:** selecciona la fila que contiene la celda seleccionada en ese momento; la fila se puede copiar y pegar en otros lugares Intercambia final: esta opción está disponible una vez hecha la selección de varias celdas. Intercambia las celdas del principio y el final de la selección actual. Incluir encabezados: lo mismo que Seleccionar fila, excepto que también se seleccionan los encabezados de la fila
- **Selección:** activa o desactiva el modo de selección
- **Tamaño de la fuente:** selecciona una fuente pequeña, mediana o de gran tamaño

0108 Vista numérica: Opciones de Trazar

ID 108 · título localizado en HelpEs

Las opciones de Trazar le ofrecen la posibilidad de ver los valores numéricos de varias de las gráficas trazadas en la Vista de gráfico. Puede seleccionar ver lo siguiente:

- Valores de los extremos
- varios puntos de interés, tales como: interceptaciones de X interceptaciones de Y Extremo horizontal Extremo vertical Puntos de inflexión

Observe que los valores que se muestran en la Vista numérica mientras se hace el trazado representan los recursos que corresponden al dominio de gráficos (es decir, entre Xmin y Xmax, y Ymin y Ymax). Por eso, si ha elegido mostrar las interceptaciones de Y de la sentencia abierta S3 y no hay interceptaciones de Y dentro del dominio de gráficos, no aparecerán valores en la columna S3.

Para que la Vista numérica vuelva a la presentación estándar, toque **[[Trazar]]** > Desactivado.

0109 Configuración numérica de Gráficos avanzados

ID 109 · título localizado en HelpEs

La Configuración numérica de la aplicación Gráficos avanzados permite configurar lo se muestra en la Vista numérica.

Campos:

- **Inicio de núm. X:** el valor inicial para el rango mostrado X
- **Inicio de núm. Y:** el valor inicial para el rango mostrado Y
- **Incr. de núm. X:** el incremento entre valores X consecutivos.
- **Incr. de núm. Y:** el incremento entre valores Y consecutivos
- **Tipo de número:** Automática = muestra los valores generados por los valores de Configuración numérica; Generar propio = especificará los valores para X e Y en la Vista numérica
- **Zoom de núm. X:** factor de zoom horizontal para operaciones de zoom
- **Zoom de núm. Y:** factor de zoom vertical para operaciones de zoom

Botones de menú:

- **[[Editar]]** permite modificar el valor seleccionado
- **[[Selec.]]** permite seleccionar una opción de un menú
- **Gráfico->:** configura los valores de Configuración numérica para coincidan con la Vista de gráfico actual

0110 Variables de aplicación Gráficos avanzados

ID 110 · título localizado en HelpEs

Para mostrar las variables relacionadas con la aplicación Gráficos avanzados, pulse Vars, toque [[Apl.]] y seleccione Gráficos avanzados.

La aplicación Gráficos avanzados incluye las variables siguientes:

- Simbólica
- Gráfico
- Numerica
- Modos

0111 Variables simbólicas de Gráficos avanzados

ID 111 · título localizado en HelpEs

Las variables simbólicas de la aplicación de Gráficos avanzados son V1 a V9 y V0.

Estas variables contienen sentencias abiertas (en X, Y, ambas o ninguna).

0112 Variables numéricas

ID 112 · título localizado en HelpEs

Esta sección enumera las variables utilizadas en la vista Numérica de la aplicación Gráficos avanzados.

0113 Variable de aplicación NumXStart

ID 113 · título recuperado de la página española

NumXStart configura el valor inicial de la variable X en la Vista numérica cuando Automática es el valor configurado para Tipo de número.

NumXStart := n, donde n es un número real

0114 Variable de aplicación NumYStart

ID 114 · título recuperado de la página española

NumYStart configura el valor inicial de la variable Y en la Vista numérica cuando Automática es el valor configurado para Tipo de número.

NumYStart := n, donde n es un número real

0115 Variable de aplicación NumXStep

ID 115 · título recuperado de la página española

NumXStep configura el tamaño de incremento (es decir, el valor de incremento) de la variable X en la Vista numérica cuando Automática es el valor configurado para Tipo de número.

NumXStep := n, donde $n > 0$

0116 Variable de aplicación NumYStep

ID 116 · título recuperado de la página española

NumYStep configura el tamaño de incremento (es decir, el valor de incremento) de la variable Y en la Vista numérica cuando Automática es el valor configurado para Tipo de número.

NumYStep := n, donde $n > 0$

0117 Variable de aplicación NumType

ID 117 · título recuperado de la página española

NumType determina cómo se generan los valores de la variable independiente en la Vista numérica:

- Automático: usa los valores de NumStart y NumStep para crear los valores de la variable independiente.
- Generar propio: permite introducir los valores de la variable independiente uno a uno.

NumType := 0 para automático (por defecto)

NumType := 1 para Generar propio

0118 Variable de aplicación NumXZoom

ID 118 · título recuperado de la página española

NumXZoom configura el factor de zoom predeterminado para hacer zoom en los valores X.

NumXZoom := n, donde n es un número real mayor que 1

0119 Variable de aplicación NumYZoom

ID 119 · título recuperado de la página española

NumYZoom establece el factor de zoom predeterminado para acercar el zoom en los valores Y.

NumYZoom := n, donde n es un número real mayor que 1

0120 Variable de aplicación NumIndep

ID 120 · título recuperado de la página española

NumIndep contiene la lista de valores que ha introducido en la Vista numérica cuando ha seleccionado el modo Generar propio.

1641 Aplicación Gráficos 3D

ID 1641 · título localizado en HelpEs

La aplicación Gráficos 3D le permite explorar las representaciones gráficas de funciones que definen a Z en términos de X e Y. En la Vista simbólica, puede definir hasta diez funciones, FZ1(X,Y) a FZ9(X,Y) y FZ0(X,Y). Resalte uno de los campos de función y comience a escribir una expresión dependiente de X e Y, o pulse la tecla de menú [[Editar]] para editar una expresión existente.

Una vez definida una función, se puede ver una tabla de sus valores o trazar su gráfico.

Para iniciar la aplicación Gráficos 3D, vaya a la biblioteca de aplicaciones y pulse el icono de la aplicación Gráficos 3D. También puede utilizar la rueda para seleccionar el icono de la aplicación Gráficos 3D, luego pulse [[Inicio]] o presione Enter para iniciar la aplicación.

1642 Vista simbólica de Gráficos 3D

ID 1642 · título localizado en HelpEs

La Vista simbólica de Gráficos 3D contiene campos para definir hasta diez funciones, cada uno define Z en términos de X e Y. Presione la tecla Symb para volver a esta vista en cualquier momento.

Los botones del menú son:

- [[Editar]]/[[Selec.]] abre una línea de entrada para editar la definición de función seleccionada o abre un cuadro de selección para realizar una selección
- [[✓]] permite activar o desactivar la definición seleccionada para la creación de gráficos y tablas
- [[X]] una ayuda de mecanografía para introducir definiciones en X
- [[Y]] una ayuda de mecanografía para introducir definiciones en Y
- [[Mostr.]] muestra la definición seleccionada en modo de pantalla completa con el desplazamiento horizontal y vertical activado
- [[Eval]] resuelve referencias a otras funciones, como $FZ2(X,Y)=FZ1(X,Y)+1$

Resalte uno de los campos de definición y comience a escribir una expresión en X e Y, o pulse [[Editar]] para abrir una línea de edición para editar una expresión existente.

1643 Vista de gráfico en Gráficos 3D

ID 1643 · título localizado en HelpEs

Presione la tecla Plot para ingresar a la Vista de gráfico de Gráficos 3D. Esta vista muestra los gráficos de las funciones definidas en la Vista simbólica. Pulse la tecla de función [[Menú]] para abrir el menú Vista de gráfico.

Los botones del menú son:

- **[[Zoom]]** abre el menú Zoom, con opciones para acercar o alejar el zoom
- **[[Trazar]]** permite activar y desactivar el cursor de trazado
- **[[Ir a]]** lleva el cursor al punto con los valores X e Y dados
- **[[Func.]]** abre el menú Func. con las siguientes opciones: Defin.: muestra la definición simbólica del gráfico actual; puede editar la expresión directamente en la Vista de gráfico
- **[[Menú]]** permite activar y desactivar el menú de la Vista de gráfico

Pulse para mover el trazador a una ubicación. Presione + para acercar la posición actual del cursor y presione - para alejarla. El menú Zoom tiene muchas de las opciones que se encuentran en el menú Zoom de la aplicación Función. Las opciones exclusivas de la aplicación Gráficos 3D se describen aquí. Todas las operaciones de zoom utilizan los factores de zoom actuales. Estos factores se establecen utilizando Factores de zoom ... en el menú Zoom. Todos los zooms descritos a continuación utilizan la ubicación actual del cursor para centrar el zoom.

Opción: Resultado

- In: Acerca las tres dimensiones
- Out: Aleja las tres dimensiones
- Z In: Acerca la dimensión Z
- Z Out: Aleja la dimensión Z
- Cuadrado XY: Hace que el rango Y sea el mismo que el rango X
- Cuadrado: Hace que el rango Y y el rango Z sean iguales a los del rango X
- Decimal: Hace que los incrementos entre los valores de ambas variables independientes sean 0.1

1644 Configuración de gráfico de Gráficos 3D

ID 1644 · título localizado en HelpEs

Presione Shift Plot para ingresar a la configuración de gráfico de Gráficos 3D. Esta vista le permite controlar la apariencia del gráfico y la Vista de gráfico. La Configuración tiene cinco páginas.

En la primera página, los campos son:

- Rng X: Los valores de X mínimo y máximo actuales que se ven en la Vista de gráfico
- Rng Y: Los valores de Y mínimo y máximo actuales que se ven en la Vista de gráfico
- Rng Z: Los valores de Z mínimo y máximo actuales que se ven en la Vista de gráfico
- Marca X: El espaciado entre las marcas en el eje X
- Marca Y: El espaciado entre las marcas en el eje Y
- Marca Z: El espaciado entre las marcas en el eje Z

Los botones del menú en la primera página son:

- **[[Editar]]** abre una línea de edición para editar el valor del campo seleccionado
- **[-Página 1/5 ▼-]** muestra la segunda página de la Configuración de gráfico (pulse en el lado derecho de la tecla de menú)

En la segunda página, los campos son:

- Cuadrícula: El número de incrementos utilizados para calcular los valores X e Y de cada gráfico
- Superficie: El campo Superficie controla el esquema de color para las gráficas. Las opciones son: Superior/Inferior: Utiliza un color para la parte superior (Z positivo, mirando hacia abajo) y otro color para la parte inferior (Z negativo, mirando para arriba) Tablero de ajedrez: Utiliza un patrón de tablero de ajedrez para colorear tanto la parte superior como la inferior de cada gráfica. También puede configurar el tamaño del cuadrado. Elevación: El color cambia según el valor Z de cada punto trazado Pendiente: El color cambia según el gradiente de cada punto trazado
- Ejes principales: Determina si la orientación de los 3 ejes se muestra o no en la parte superior izquierda de la Vista de gráfico. Si está marcada, también puede configurar los colores para los ejes.

Las teclas de menú son:

- **[[Editar]]/[[Selec.]]** abre una línea de entrada para editar el campo seleccionada o abre un cuadro de selección para realizar una selección
- **[[✓]]** activa o desactiva la selección

En la tercera página de la Configuración de gráfico, los campos son:

- Lados del cuadro Ninguno: no colorear ninguna de las caras del marco del cuadro Posterior: Colorea cada una de las caras X-Y, Y-Z, y X-Z del marco del cuadro que se ubican detrás de los gráficos Zmin: Colorea la cara X-Y en Zmin
- Marco del cuadro Ninguna: No dibuja el marco del cuadro alrededor de los gráficos Posterior: Dibuja solo los 9 segmentos del marco del cuadro que están detrás de los gráficos Frontal y posterior: Dibuja el marco completo del cuadro
- Ejes del cuadro Ninguna: No dibuja los ejes Posterior: Dibuja los ejes detrás de los gráficos Frontal y posterior: Dibuja los ejes delante y detrás de los gráficos
- Líneas del cuadro Ninguna: No dibuja las líneas de cuadrícula de la marca Posterior: Dibuja las líneas de cuadrícula de la marca que se sitúan detrás de los gráficos Frontal y posterior: Dibujar las líneas de cuadrícula de marca tanto detrás como delante de los gráficos
- Puntos del cuadro Ninguna: No dibuja los puntos de cuadrícula de la marca Posterior: Dibuja los puntos de cuadrícula de la marca que se sitúan detrás de los gráficos Frontal y posterior: Dibujar los puntos de cuadrícula de marca tanto detrás como delante de los gráficos
- Etiquetas: Marca para etiquetar los ejes

Las teclas de menú aquí son las mismas que en la página 2.

Las opciones de Configuración de gráfico en la página 4 son:

- Cursor Estándar: Cursor blanco constante Inversión Parpadeo: Cursor blanco parpadeante
- Escala del cuadro: El factor de escala para el marco del cuadro. 0.5-2. El valor predeterminado es 1.
- Representar eje X Un número real: La coordenada X del punto final del vector de rotación
- Representar eje Y Un número real: La coordenada Y del punto final del vector de rotación
- Representar eje Z Un número real: La coordenada Z del punto final del vector de rotación
- Representar Giro Un número real: El ángulo de rotación (en radianes) del vector de rotación

Las teclas de menú son las mismas que en las páginas 2 y 3.

La última página de la Configuración de gráfico es igual a la de la página 3 de la Configuración de gráfico para las aplicaciones Función, Polar y Paramétricas. Aquí se selecciona una imagen que se utilizará como fondo en la Vista de gráfico.

1645 Vista numérica de Gráficos 3D

ID 1645 · título localizado en HelpEs

Presione la tecla Num para ingresar a la Vista numérica. La Vista numérica de Gráficos 3D está diseñada para crear y explorar una tabla de valores X/Y/Z, que se basa en las funciones definidas en la Vista simbólica.

Pulse cualquier fila de las columnas X o Y, ingrese cualquier valor real y pulse [[OK]]. La tabla se reconfigurará. También puede desplazar la tabla arrastrándola o moviéndola con un dedo.

Los botones del menú son:

- [[Zoom]] abre el menú Zoom. Tenga en cuenta que en la Vista numérica, el zoom cambia el incremento entre los valores de X e Y consecutivos. Acercar el zoom disminuye el incremento, alejarlo lo aumenta. Los valores de la fila que se acercan o se alejan permanecen iguales.
- [[Más]] abre el menú Más, idéntico al de la aplicación Función o Creación de gráficos avanzados
- [[Ir a]] salta a los valores especificados de las variables independientes X e Y
- [[Defn]] muestra la definición de la columna seleccionada

El menú [[Zoom]] de la Vista numérica tiene muchas de las mismas opciones que el menú Zoom de la Vista numérica de las otras aplicaciones gráficas. Estas son las operaciones de zoom exclusivas de la aplicación Gráficos 3D. Todas las operaciones de zoom utilizan los factores de zoom actuales. Estos factores se establecen en la Configuración numérica.

Opción: Resultado

- In: Acerca la fila actual de la tabla tanto en las columnas X como Y
- Out: Aleja la fila actual tanto en las columnas X como Y
- Y In: Acerca la fila actual solo en la columna Y
- Y Out: Aleja la fila actual solo en la columna Y
- Decimal: Hace que los incrementos entre los valores de ambas variables independientes sean 0.1
- Entero: Hace que los incrementos entre los valores de ambas variables independientes sean 1
- Disp.: Hace que el incremento entre los valores de ambas variables independientes sea $\pi/24$ (radianes)

- Un-Zoom: Deshace el último zoom

1646 Configuración numérica de Gráficos 3D

ID 1646 · título localizado en HelpEs

La Configuración numérica de Gráficos 3D extiende la Configuración numérica de Función para cubrir tanto las variables independientes X como Y en lugar de solo X. La configuración numérica de Gráficos 3D le permite establecer valores de inicio y de incremento para X e Y, así como establecer los factores de zoom tanto para X como para Y.

1647 Variables de la aplicación Gráficos 3D

ID 1647 · título localizado en HelpEs

Para mostrar las variables relacionadas con la aplicación Gráficos 3D, pulse Vars, pulse [[Apl.]] y seleccione Gráficos 3D.

1669 Variables simbólicas

ID 1669 · título localizado en HelpEs

Las variables simbólicas en la aplicación Gráficos 3D son FZ1(X, Y) a FZ9(X, Y) y FZ0(X, Y). Cada una de las variables contiene una expresión que define Z en términos de X e Y.

1673 Variables de gráfico

ID 1673 · título localizado en HelpEs

Las variables de gráficos Gráficos 3D incluyen muchas de las variables de gráfico de las otras aplicaciones gráficas como Función y Gráficos avanzados. Esta sección incluye solo aquellas variables de gráfico exclusivas de la aplicación Gráficos 3D.

1674 Variables, Página 1

ID 1674 · título localizado en HelpEs

Las variables de aplicación en la Vista de gráfico exclusivas en la página 1 de la Configuración de gráfico son:

- Zmin: el valor Z mínimo
- Zmax: el valor Z máximo
- Ztick: el espaciado de la marca en el eje Z

1675 Variables, Página 2

ID 1675 · título localizado en HelpEs

Las variables de aplicación en la Vista de gráfico exclusivas en la página 2 de la Configuración de gráfico son:

- GridX: el número de pasos utilizados de Xmin a Xmax para trazar puntos
- GridY: el número de pasos utilizados de Ymin a Ymax para trazar puntos
- Superficie: contiene una lista que define el esquema de color {0}: Arriba/Abajo {1, c, c}: Tablero de ajedrez; cada cuadrado es c por c {2}: Elevación {3}: Pendiente
- KeyAxes: contiene 0 si los ejes principales no están marcados y 1 si lo están

1676 Variables, Página 3

ID 1676 · título localizado en HelpEs

Las variables de aplicación en la Vista de gráfico exclusivas en la página 3 de la Configuración de gráfico son:

- BoxSides: qué caras del marco del cuadro están coloreadas 0: Ninguno - no colorear ninguna de las caras del marco del cuadro 1: Posterior - colorear las tres caras del marco del cuadro que se encuentran detrás del o los gráficos 2: Zmin - colorear la cara que se encuentra en Zmin
- BoxFrame: cómo se dibuja el marco del cuadro 0: Ninguno - no se muestra el marco del cuadro 1: Posterior - muestra las tres caras del marco del cuadro que se encuentran detrás del o los gráficos 2: Frontal y posterior: muestra todas las 6 caras del marco del cuadro

- **BoxAxes:** cómo se dibujan los tres ejes 0: Ninguno - no se muestran los ejes 1: Posterior - muestra los tres ejes que se encuentran detrás del o los gráficos 2: Frontal y posterior - muestra los ejes delante y detrás del o los gráficos
- **BoxLines:** cómo se dibujan las líneas de cuadrícula en el marco del cuadro 0: Ninguno - no se dibujan líneas de cuadrícula en el marco del cuadro 1: Posterior - dibuja líneas de cuadrícula en las tres caras del marco del cuadro que se encuentran detrás del o los gráficos 2: Frontal y posterior - dibuja las líneas de cuadrícula en todas las caras del bastidor de la caja
- **BoxDots:** cómo se dibujan los puntos de cuadrícula en el marco del cuadro 0: Ninguno - no se dibujan puntos de cuadrícula en el marco del cuadro 1: Posterior - dibuja puntos de cuadrícula en las tres caras del marco del cuadro que se encuentran detrás del o los gráficos 2: Frontal y posterior - dibuja los puntos de cuadrícula en todas las caras del bastidor de la caja
- **BoxScale:** un número real de 0,5 a 2 que determina el factor de escala utilizado para dibujar el marco del cuadro

1677 Variables, Página 4

ID 1677 · título localizado en HelpEs

Las variables de aplicación en la Vista de gráfico exclusivas en la página 4 de la Configuración de gráfico son:

- **BoxScale:** un número real entre 0,5 y 2 que determina la escala para el marco del cuadro
- **PoseXaxis:** la coordenada X del punto final del vector de rotación
- **PoseYaxis:** la coordenada Y del punto final del vector de rotación
- **PoseZaxis:** la coordenada Z del punto final del vector de rotación
- **PoseTurn:** el ángulo de rotación (en radianes) del eje de representación

1678 Variables, Página 5

ID 1678 · título localizado en HelpEs

No hay variables de aplicación en la Vista de gráfico que pertenezcan únicamente a la aplicación Gráficos 3D en la página 5 de la Configuración de gráfico. Solamente se usan las tres variables comunes de la aplicación de Vista de gráfico, **ImageName**, **ImageDisplay** e **ImageOpacity**.

1679 Variables numéricas

ID 1679 · título localizado en HelpEs

Hay tres variables de aplicación de la Vista numérica exclusivas de la aplicación Gráficos 3D:

- **NumYStart:** el valor inicial para Y en la Vista numérica
- **NumYStep:** el valor de incremento para Y en la Vista numérica
- **NumYZoom:** el valor de zoom para Y en la Vista numérica

0121 Aplicación Geometría

ID 121 · título localizado en HelpEs

La aplicación Geometría permite crear, editar y explorar construcciones geométricas. Una construcción geométrica puede estar compuesta por un número diverso de objetos geométricos, como puntos, líneas, polígonos, curvas, tangentes, etc. Puede tomar mediciones (como áreas y distancias), manipular objetos y anotar cómo cambian las mediciones a medida que manipula la construcción.

Pulse **[[Inicio]]** o presione **Enter** para iniciar la aplicación. La aplicación se abre en la Vista de gráfico.

0122 Vista de gráfico de Geometría

ID 122 · título localizado en HelpEs

En la Vista de gráfico puede dibujar objetos directamente en la pantalla mediante varias herramientas de dibujo. Por ejemplo, para dibujar un círculo, toque **[[Cmds]]** y seleccione **Círculo**. Toque ahora el lugar en el que le gustaría que estuviera el centro del círculo y pulse **Enter**. A continuación, toque un punto que vaya a estar en la circunferencia y pulse **Enter**. Se dibujará un círculo cuyo centro estará en la ubicación que tocó por primera vez, con un radio igual a la distancia entre el primer y el segundo toque.

La selección de un objeto normalmente implica dos pasos: pulsar para seleccionar una ubicación u objeto y, a continuación, presionar **Enter**. A menudo, habrá varios objetos cerca de donde pulse la pantalla; aparecerá un

menú emergente que le permitirá seleccionar el o los objetos que desee. Marque los que desea seleccionar y pulse [[OK]]. De lo contrario, presione Enter para confirmar su intención de crear el punto o seleccionar el objeto. Al crear un punto, puede pulsar la pantalla y luego usar la rueda para posicionar con precisión el punto antes de presionar Enter.

La parte inferior de la pantalla siempre contiene ayuda para guiarlo. Podrá ver cualquiera o todos los siguientes:

- Si la herramienta está activa, verá la ayuda en la parte inferior izquierda (por encima de la tecla de menú [[Cmds]])
- Si una herramienta está activa, verá el comando actual ya que quedará registrado en la Vista simbólica a la derecha del texto de ayuda
- las coordenadas del puntero actual se muestran en la parte inferior izquierda
- los objetos reconocidos en las coordenadas del puntero se muestran en la parte inferior derecha. Si presiona Enter, estos objetos aparecerán en el menú emergente

Botones de menú:

- [[Cmds]] El menú de comandos tiene las siguientes categorías: Zoom: muestra un menú de opciones de zoom para acercar o alejar la Vista de gráfico Punto: muestra un menú para crear varios tipos de puntos Línea: muestra un menú para crear varios tipos de objetos rectos (segmentos, líneas, tangentes, etc.) Polígono: muestra un menú para crear varios tipos de polígonos Curva: muestra un menú para dibujar secciones cónicas, etc. Gráfico: muestra un menú para graficar funciones, etc. Transformar: muestra un menú que ofrece varios tipos de transformaciones (reflexión, rotación, etc.) Cartesiano: muestra un menú para presentar coordenadas de puntos, ecuaciones cartesianas y así por el estilo Medida: muestra un menú de varias medidas, tales como distancia, pendiente, etc. Pruebas: muestra un menú de varias pruebas que puede realizar, como is_collinear (prueba si un conjunto de puntos es colineal o no)

PROCESO

Elija una herramienta de uno de los menús enumerados antes y siga las instrucciones de la pantalla. Presione Enter después de seguir cada instrucción y luego presione Esc para desactivar la herramienta de dibujo.

Cada objeto geométrico que se crea en la Vista de gráfico tiene su propia entrada o entradas en la Vista simbólica. De hecho, puede crear objetos directamente en la Vista simbólica y aparecerán en la Vista de gráfico. puede crear la mayoría de los objetos geométricos en cualquier vista. El menú [[Cmds]] aparece en ambas vistas.

Los cálculos (mediciones, pruebas, etc.) creados desde los menús Cartesiano, Medición o Pruebas aparecen en la parte superior de la pantalla y se anclan ahí, es decir, no salen de su lugar aunque usted recorra o haga un zoom a la pantalla. Puede editar las etiquetas predeterminadas para estos objetos. Toque y sostenga y aparecerá una línea de edición para que introduzca su propia etiqueta descriptiva. Toque [[OK]] cuando haya terminado. Puede seleccionar y mover esos objetos para desanclarlos. Una vez que los haya desanclado, se mueven conforme se desplaza y hace zoom en la Vista de gráfico. Para volver a anclar uno de estos objetos, toque y sostenga. La línea de edición volverá a aparecer. Note que la tecla de menú [[Dock]] está a la izquierda. Toque [[Dock]] y el objeto volverá a su posición anclada. Cada cálculo que crea tiene su propia entrada en la Vista numérica, tal como cada objeto geométrico que usted crea tiene su propia entrada en la Vista simbólica. También puede crear cálculos y sus etiquetas en la Vista numérica.

Una vez que haya creado objetos en la Vista de gráfico, puede seleccionar y moverlos, uno a la vez o como un grupo. Para seleccionar un grupo de objetos, pulse y mantenga presionado, luego arrastre para crear un rectángulo de selección. Todo lo que quede dentro del rectángulo se selecciona. Arrastre para mover los objetos seleccionados y pulse Esc o Enter cuando termine.

MENÚ DE OPCIONES

Cuando se selecciona un objeto, aparece una nueva tecla del menú [[Opciones]]. Pulse esta tecla para seleccionar entre las opciones para el objeto seleccionado, incluyendo las opciones de color y relleno. Las opciones varían, dependiendo del tipo de objeto seleccionado, pero incluyen las siguientes:

- Selec. color: abre el selector de color para seleccionar un color para el objeto
- Relleno: alterna entre el objeto relleno con su color actual y sin relleno
- Trazo: alterna entre trazo y sin trazo para un punto seleccionado
- Animar: inicia y detiene una animación basada en el objeto seleccionado
- Punto de estilo: abre un cuadro de selección para seleccionar un estilo para el punto seleccionado

Puede hacer el gesto de pinza (ponga dos dedos sobre la pantalla y júntelos o sepárelos) para ampliar o disminuir la imagen. Puede arrastrarse a la Vista de gráfico. Por supuesto, + y - sirven para hacer zoom y ampliar o disminuir, respectivamente, en el puntero.

0123 Teclas de método abreviado

Las teclas de método abreviado del teclado funcionan en la Vista de gráfico únicamente. Las teclas de método abreviado del teclado se ordenan alfabéticamente; Sin embargo, no tiene que pulsar la tecla ALPHA primero, simplemente pulse la tecla en la que aparece la letra correspondiente.

Las teclas de método abreviado son:

- A: activa o desactiva los ejes (presione Vars)
- C: crear un nuevo círculo (oprima la tecla Plantilla)
- I: encuentra una intersección de dos objetos (presione TAN)
- L: crea una nueva línea (presione x^2)
- P: crea un nuevo punto (presione EEX)
- S: crea un nuevo segmento (presione 9)
- T: crea un nuevo triángulo (presione $\div$)

0124 Configuración de gráfico de Geometría

La vista Config. de gráfico de la aplicación Geometría permite controlar la apariencia de la Vista de gráfico, incluidos el rango de los valores que se muestran, si los ejes se dibujan o no, etc.

Campos:

- Rng X: el rango horizontal
- Rng Y: el rango vertical
- Tamaño de píxel: el tamaño de cada píxel cuadrado en unidades cartesianas
- Mrc X: Espaciado horizontal entre marcas
- Mrc Y: Espaciado vertical entre marcas
- Ejes: muestra u oculta las etiquetas correspondientes a los ejes
- Etiquetas: muestra u oculta las etiquetas (X, Y) y/o flechas para cada eje
- Puntos cuadr.: muestra u oculta los puntos de una cuadrícula
- Líneas cuadríc.: muestra u oculta las líneas de una cuadrícula
- Despl. texto: desplazamiento automático del texto actual del comando en el área de ayuda
- Coordenadas: muestra u oculta las coordenadas en los ejes

La Vista de gráficos de la aplicación Geometría siempre mantiene una ventana cuadrada; es decir, una en la que las unidades horizontales y verticales siguen siendo las mismas. Por esta razón, no se puede modificar el rango X e Y tan libremente como puede hacerlo en las otras aplicaciones.

Puede establecer el valor X mínimo y el valor Y mínimo y, a continuación, establecer el tamaño de un píxel. Si ajusta el tamaño del píxel a 0.1 cambiará las coordenadas del puntero por 0.1 cada vez que presione la rueda (arriba, abajo, izquierda o derecha). Al hacer elecciones en estos tres campos, los otros dos se calculan y se muestran automáticamente.

0125 Variable ScrollText

ScrollText controla la propiedad de Recorrer texto en la pantalla de Configuración de gráfico de Geometría.

ScrollText := 0 se elimina la marca de la casilla de verificación de Recorrer texto (opción predeterminada)

ScrollText := 1 se marca la casilla de verificación de Recorrer texto.

Cuando ScrollText=1, se recorrerá un texto largo en el menú de la Vista de gráfico de Geometría.

0126 Vista simbólica de Geometría

Cada objeto (ya sea un punto, un segmento, una línea, un polígono o una curva) recibe un nombre y su definición se muestra en la Vista simbólica. El nombre es el nombre que recibe el objeto y que puede ver en la Vista de gráfico con "G" como prefijo. Por lo tanto, un punto con la etiqueta A en la Vista de gráfico recibe el nombre de GA en la Vista simbólica.

En la Vista simbólica, puede modificar los objetos existentes y crear nuevos objetos. Cada definición del objeto aparece en su propia línea y tiene las siguientes partes:

- Casilla de verificación: la casilla de verificación determina si el objeto definido se muestra u oculta en la Vista de gráfico. Marque para mostrar y desmarque para ocultar el objeto
- Cuadro de color: toque en el cuadro de color para abrir el selector de color. Toque un color para seleccionarlo y cerrar el selector
- Campo Nombre: el campo nombre determina un nombre único para el objeto. Normalmente, el nombre se pondrá automáticamente.
- Campo Definición: el campo de la definición es donde se introduce el comando para definir el objeto simbólicamente. Abra el menú Cmds y elija un comando

Puede mover hacia arriba o abajo para desplazarse por la pantalla cinéticamente.

Botones de menú: VISTA PRINCIPAL

- [[Editar]] permite modificar una definición seleccionada
- [[✓]] permite seleccionar o anular la selección de un objeto. Sólo los objetos seleccionados aparecen en la Vista de gráfico.
- [[Nuevo]] toque este botón para crear un objeto nuevo. Los botones cambian (consulte a continuación).
- [[↓]] permite mover el objeto seleccionado hacia abajo en una lista de objetos
- [[↑]] permite mover el objeto seleccionado hacia arriba en una lista de objetos
- [[Elimi.]] permite eliminar el objeto seleccionado

Botones de menú: VISTA CREACIÓN DE OBJETO

Si toca [[Nuevo]] en la Vista simbólica principal (ver arriba), los botones que aparecen en la parte inferior de la pantalla cambian a:

- [[Cmds]] el menú comandos tiene muchas de las mismas categorías que la Vista de gráfico Punto: muestra un menú para crear varios tipos de puntos Línea: muestra un menú para crear varios tipos de objetos rectos (segmentos, líneas, tangentes, etc.) Polígono: muestra un menú para crear varios tipos de polígonos Curva: muestra un menú para dibujar secciones cónicas, etc. Gráfico: muestra un menú para graficar funciones, etc. Transformar: muestra un menú que ofrece varios tipos de transformaciones (reflexión, rotación, etc.)

Hay otros comandos geométricos disponibles en el catálogo. Pulse la tecla de Cuadro de herramientas, pulse la tecla menú [[Ctlg]] y desplácese por la lista. Puede pulsar cualquier combinación de dos letras para saltar a la ubicación aproximada de un comando. Consulte la ayuda de las "Funciones de Geometría" para obtener más detalles sobre todos los comandos geométricos.

Otros botones de menú: VISTA CREACIÓN DE OBJETO

- [[Vars.]]
- [[x]] permite introducir una x
- [[y]] permite introducir una y
- [[Canc.]] permite cancelar la operación.
- [[OK]] permite finalizar la operación

Observe que la aplicación Geometría usa la x y la y en minúscula para definir gráficos y hacer cálculos.

0127 Vista numérica de Geometría

ID 127 · título localizado en HelpEs

La Vista numérica le permite crear cálculos y realizar pruebas en la aplicación Geometría. Los resultados mostrados son dinámicos: si manipula un objeto en la Vista de gráfico o la Vista simbólica, cualquier cálculo de la Vista numérica que se refiera a ese objeto se actualiza automáticamente para reflejar las nuevas propiedades numéricas del objeto. Como sucede en la Vista simbólica, puede marcar una entrada para hacerla visible en la Vista de gráfico o desmarcarla para ocultarla en la Vista de gráfico.

Botones de menú: VISTA PRINCIPAL

- [[Editar]] permite modificar una medida o prueba seleccionada.
- [[✓]] permite seleccionar o deseleccionar una medida o prueba. Las medidas y pruebas seleccionadas aparecen en la Vista de gráfico.
- [[Nuevo]] toque este botón para crear una medida o prueba nueva. Los botones cambian (consulte a continuación).
- [[↓]] permite mover la medida o prueba seleccionada hacia abajo en la lista.

- **[[↑]]** permite mover la medida o prueba seleccionada hacia arriba en la lista.
- **[[Elim.]]** permite borrar la medida o prueba seleccionada.

Botones de menú: VISTA CREACIÓN DE OBJETO

Si toca **[[Nuevo]]** en la Vista numérica principal (ver arriba), los botones que aparecen en la parte inferior de la pantalla cambian a:

- **[[Cmds]]** muestra un menú de todas las medidas y pruebas específicas de geometría.
- **[[Vars.]]** muestra un menú de todos los objetos definidos en la Vista simbólica.
- **[[Canc.]]** permite cancelar la operación.
- **[[OK]]** permite finalizar la operación.

MENÚ DE COMANDOS DE LA VISTA NUMÉRICA

- **Cartesiano:** muestra un menú para presentar coordenadas de puntos, ecuaciones cartesianas, etc.
- **Medida:** muestra un menú de varias medidas, tales como distancia, pendiente, etc.
- **Pruebas:** muestra un menú de varias pruebas que puede realizar, como **is_collinear**, que prueba si un conjunto de puntos es colineal

Además de crear pruebas y medidas, puede crear cálculos basados en resultados de pruebas y medidas. Solo tiene que tocar **[[Nuevo]]** e introducir el cálculo del mismo modo que lo haría en las vistas del sistema algebraico computacional o de inicio. Cada cálculo que define en la Vista numérica aparece en su propia línea y tiene las siguientes partes:

- **Casilla de verificación:** la casilla de verificación determina si el objeto definido se muestra u oculta en la Vista de gráfico. Márquela para mostrar el objeto y desmárquela para ocultarlo.
- **Cuadro de color:** toque el cuadro de color para abrir el selector de color. Toque un color para seleccionarlo y cerrar el selector; la etiqueta de texto y el resultado numérico aparecerán en la Vista de gráfico en el color seleccionado.
- **Campo de etiqueta:** el campo de etiqueta contiene la etiqueta predeterminada para su cálculo. Toque Etiqueta para cambiar la etiqueta.
- **Campo de definición:** el campo de definición es donde se introduce el comando que define su cálculo. Abra el menú **Cmds** y elija un comando. También puede introducir funciones directamente desde el teclado.

Observe que la aplicación Geometría usa las letras **x** e **y** en minúscula para definir gráficos y realizar cálculos.

0128 Variables de Geometría

ID 128 · título localizado en HelpEs

Además de las variables de modos y de la Vista de gráfico (que son comunes a todas las aplicaciones), la aplicación Geometría tiene las siguientes variables adicionales en la Vista de gráfico:

- **Tam. Píxel:** determina el tamaño de cada píxel cuadrado en unidades cartesianas
- **Depl. texto:** determina si el actual texto del comando en la Vista de gráfico se desplaza automáticamente o manualmente. **Depl. texto := 0** desplazamiento manual (predeterminado) **Depl. texto := 1** desplazamiento automático

Las otras variables de la aplicación Geometría se explican en Aplicaciones de HP > Variables de aplicaciones comunes > Variables comunes de la Vista de gráfico.

La aplicación Geometría también enumera las variables de las aplicaciones de cada objeto en la Vista simbólica.

0129 Funciones de geometría

ID 129 · título localizado en HelpEs

Este menú contiene todos los comandos y las funciones específicas de la geometría.

0130 Punto

ID 130 · título localizado en HelpEs

Este menú contiene comandos de opciones para crear puntos, puntos medios y así sucesivamente.

0131 Punto

ID 131 · título recuperado de la página española

Sintaxis

```
point(Real1, Real2 [, Real3])  
point(Expr1, Expr2 [, Expr3])
```

Crea un punto, dadas las coordenadas del punto. Cada coordenada puede ser un valor o una expresión que involucra variables o medidas sobre otros objetos en la construcción geométrica.

Ejemplos

[Ejemplo gráfico E1] crea un punto cuyas coordenadas son (3,4). Este punto puede seleccionarse y moverse más adelante.
[Ejemplo gráfico E2] crea un punto cuya coordenada x es igual a la de un punto A y cuya coordenada y es igual a la del un punto B. Este punto va a cambiar para reflejar los movimientos del punto A o punto B.
[Ejemplo gráfico E3]

0132 Punto sobre

ID 132 · título recuperado de la página española

Sintaxis

```
element(Object, Real)  
element(Real1..Real2)
```

Crea un punto sobre un objeto geométrico cuyo parámetro, por ejemplo la abscisa, tiene un valor determinado.

También puede crear un control deslizante: un valor real en un intervalo determinado representado como una barra deslizante.

El valor creado mediante `element(Real1..Real2)` puede utilizarse como coeficiente en una función que se defina posteriormente en la Vista simbólica y se trace en la Vista de gráfico de la aplicación Geometry. También puede utilizarse en una medición o un cálculo en la Vista numérica.

Para abrir y manipular un control deslizante, toque y mantenga pulsado su nombre. Utilice la tecla de menú [[Editar]] para definir el control deslizante con mayor precisión, crear animaciones y acceder a opciones relacionadas. Pulse Esc para cerrar la barra deslizante con el nuevo valor, o toque cualquier otra parte de la pantalla.

Ejemplos

[Ejemplo gráfico E1] Crea un punto en la gráfica de $y = x^2$. Inicialmente, este punto aparece en (-2, 4). Puede mover el punto, pero siempre permanece en la gráfica de la función.
[Ejemplo gráfico E2] Crea una barra deslizante con un valor inicial de 2,5.

0133 Punto medio

ID 133 · título recuperado de la página española

Sintaxis

```
midpoint(Segmento)  
midpoint(Punto1, Punto2)
```

Devuelve el punto medio de un segmento. El argumento puede ser el nombre de un segmento o dos puntos que definen un segmento. En este último caso, el segmento no necesita ser dibujado.

0134 Centro

ID 134 · título recuperado de la página española

Sintaxis

```
center(Círculo)
```

Devuelve el centro de un círculo. El círculo se puede definir por el comando círculo o por el nombre (por ej., GC).

0135 Intersección única

ID 135 · título recuperado de la página española

Sintaxis

```
single_inter(Curva1, Curva2, [Punto])
```

Devuelve la intersección de Curva1 y Curva2 que está más cerca de Punto.

En la Vista de gráfico, este comando le pedirá que seleccione dos curvas. A continuación, aparecerá un punto; muévelo hasta la intersección deseada y presione Enter. Puede mover el punto más adelante para cambiar las

intersecciones si lo desea.

0136 Punto aleatorio

ID 136 · título localizado en HelpEs

Crea un punto aleatorio en la Vista de gráfico.

Activar este comando y presione Enter para crear un punto aleatorio en la Vista de gráfico. Presione Enter para crear puntos más aleatorios o pulse Esc para salir.

0137 Intersecciones

ID 137 · título recuperado de la página española

Sintaxis

`inter(Curva1, Curva2)`

Devuelve la intersección de dos curvas como un vector.

Ejemplos

[Ejemplo gráfico E1], indica que hay dos intersecciones-una en (6,2) y la otra en (-9,-11/2).

0138 Trazar

ID 138 · título recuperado de la página española

Sintaxis

`trace(point)`

Trazar es una opción en el menú de opciones de la Vista de gráfico de la aplicación de Geometría. Es un palanca para activar y desactivar el seguimiento para el punto seleccionado. También hay una opción para borrar un trazo existente de la Vista de gráfico.

0139 Eliminar trazo

ID 139 · título localizado en HelpEs

Borra el trazo actual en la Vista de gráfico. No detiene el trazado posterior.

0140 Línea

ID 140 · título localizado en HelpEs

Este menú contiene todas las funciones geométricas específicas a los objetos rectos (segmentos, líneas, etc.).

0141 Segmento

ID 141 · título recuperado de la página española

Sintaxis

`segment(Punto1, Punto2)`

Dibuja un segmento definido por sus puntos extremos.

Ejemplos

[Ejemplo gráfico E1] dibuja el segmento definido por los puntos con coordenadas (1, 2) y (4, 0).
[Ejemplo gráfico E2] dibuja el segmento AB.
[Ejemplo gráfico E3]

0142 Raya

ID 142 · título recuperado de la página española

Sintaxis

`half_line(Punto1, Punto2)`

Dados 2 puntos, dibuja una raya desde el primer punto a través del segundo punto.

Ejemplos

[Ejemplo gráfico E1] dibuja una raya desde el origen que pasa por el punto en (1,1)
[Ejemplo gráfico E2]

0143 Línea

ID 143 · título recuperado de la página española

Sintaxis

```
line(Punto1, Punto2)
line(a*x+b*y+C)
line(punto1, slope=realm)
```

Dibuja una línea en la Vista de gráfico de la aplicación de geometría o devuelve la ecuación de una línea en la Vista CAS. Los argumentos pueden ser dos puntos, una expresión lineal con la forma $a*x+b*y+c$ o un punto y una pendiente.

Ejemplos

[Ejemplo gráfico E1] dibuja la línea cuya ecuación es $y=x-1$; es decir, la línea a través de los puntos (2,1) y (3,2).
[Ejemplo gráfico E2] dibuja la línea cuya ecuación es $2x-3y=8$
[Ejemplo gráfico E3] dibuja la línea cuya ecuación es $x-2y=7$; es decir, la línea a través de (3, -2) con pendiente $m=1/2$
[Ejemplo gráfico E4]
[Ejemplo gráfico E5]

0144 Paralelo

ID 144 · título recuperado de la página española

Sintaxis

```
parallel(Punto, Línea)
```

Dados un punto y una línea, devuelve la ecuación de la línea mediante el punto paralelo a la línea dada.

Ejemplos

[Ejemplo gráfico E1] dibuja la línea a través del punto A paralelo a la línea B.
[Ejemplo gráfico E2] dibuja la línea a través del punto (3, -2) paralelo a la línea cuya ecuación es $x+y=5$; es decir, la línea cuya ecuación es $y=-x+1$.
[Ejemplo gráfico E3]

0145 Perpendicular

ID 145 · título recuperado de la página española

Sintaxis

```
perpendicular(Punto, Línea)
perpendicular(Punto1, Punto2, Punto3)
```

Dibuja una línea que pasa a través de un punto dado perpendicular a una línea dada. La línea puede definirse por su nombre, dos puntos o una expresión en x e y .

Ejemplos

[Ejemplo gráfico E1] dibuja una línea perpendicular a la línea D que pasa a través del punto A.
[Ejemplo gráfico E2] dibuja una línea que pasa a través del punto cuyas coordenadas son (3, 2) que es perpendicular a la línea BC.
[Ejemplo gráfico E3] dibuja una línea que pasa a través del punto cuyas coordenadas son (3, 2), que es perpendicular a la línea cuya ecuación es $x - y = 1$; es decir, la línea cuya ecuación es $y=-x+5$.
[Ejemplo gráfico E4]

0146 Tangente

ID 146 · título recuperado de la página española

Sintaxis

```
tangent(Curva, Punto)
```

Dibuja la/las tangentes a una curva dada a través de un punto dado. El punto no tiene que ser un punto de la curva.

Ejemplos

[Ejemplo gráfico E1] dibuja la tangente al gráfico $y=x^2$ a través del punto (1,1); es decir, la línea cuya ecuación es $y=2*x-1$.

[Ejemplo gráfico E2] dibuja la tangente al gráfico de $y=x^2$ a través del punto A. Luego, puede mover el punto A y la tangente se moverá con él.

[Ejemplo gráfico E3] dibuja una o más líneas tangentes a través del punto A al círculo cuyo centro está en el punto B y cuyo radio se define por el segmento BC.

[Ejemplo gráfico E4]

0147 Mediana

ID 147 · título recuperado de la página española

Sintaxis

`median_line(Punto1, Punto2, Punto3)`

Dados tres puntos que definen un triángulo, dibuja la mediana del triángulo que atraviesa el primer punto y que contiene el punto medio del segmento definido por los otros dos puntos. En la Vista CAS, devuelve la ecuación de la línea media.

Ejemplos

[Ejemplo gráfico E1] dibuja la línea cuya ecuación es $y=2x$; es decir, la línea a través del primer vértice del triángulo en (0,0) y el punto en (2,4), el punto medio del segmento cuyos extremos son (0, 8) y (4, 0).

[Ejemplo gráfico E2]

0894 altitude — Dados tres puntos no lineales, dibuja altitude del triángulo definido por los tres puntos que pasan a...

ID 894 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: altitude

Explicación española: Dados tres puntos no lineales, dibuja altitude del triángulo definido por los tres puntos que pasan a...

Sintaxis

`altitude(punto1, punto2, punto3)`

Dados tres puntos no lineales, dibuja altitude del triángulo definido por los tres puntos que pasan a través del primer punto. No es necesario dibujar el triángulo.

Ejemplos

[Ejemplo gráfico E1] dibuja una línea que pasa a través del punto (6,6) perpendicular a la línea que pasa a través de ambos puntos (-2,3) y (5,1).

[Ejemplo gráfico E2]

0148 Bisector

ID 148 · título recuperado de la página española

Sintaxis

`bisector(Punto1, Punto2, Punto3)`

Dados tres puntos, crea el bisector del ángulo definido por los tres puntos cuyo vértice se encuentra en el primer punto. No es necesario dibujar el ángulo en la Vista de gráfico.

Ejemplos

[Ejemplo gráfico E1]

[Ejemplo gráfico E2]

[Ejemplo gráfico E3] dibuja el bisector de $\angle BAC$.

[Ejemplo gráfico E4] dibuja la línea dada por $y=-x$

0149 Polígono

ID 149 · título localizado en HelpEs

Este menú contiene todas las funciones geométricas específicas a los polígonos.

0150 Triángulo

ID 150 · título recuperado de la página española

Sintaxis

```
triangle(Point1, Point2, Point3)
```

Dibuja un triángulo a partir de sus tres vértices.

Ejemplos

[Ejemplo gráfico E1]
[Ejemplo gráfico E2] dibuja $\triangle ABC$
[Ejemplo gráfico E3]

0151 Triángulo isósceles

ID 151 · título recuperado de la página española

Sintaxis

```
isosceles_triangle(Punto1, Punto2, Ángulo [, Var])
```

Dibuja un triángulo isósceles definido por dos de sus vértices y un ángulo. Los vértices definen uno de los dos lados con la misma longitud y el ángulo define el ángulo entre los dos lados con la misma longitud. Al igual que `equilateral_triangle`, tiene la posibilidad de almacenar las coordenadas del tercer punto en una variable del sistema algebraico computacional.

Ejemplos

[Ejemplo gráfico E1]
[Ejemplo gráfico E2] define un triángulo isósceles de modo que uno de los dos lados con la misma longitud es AB y el ángulo entre los dos lados con la misma longitud tiene una medida igual a la de $\angle ACB$.

0152 Triángulo rectángulo

ID 152 · título recuperado de la página española

Sintaxis

```
right_triangle(Punto1, Punto2, Realk)
```

Dibuja un triángulo rectángulo dados dos puntos y un factor de escala. Un lado del triángulo rectángulo está definido por los dos puntos, el vértice del ángulo recto se encuentra en el primer punto, y el factor de escala multiplica la longitud del primer lado para determinar la longitud del segundo.

Ejemplos

[Ejemplo gráfico E1]
[Ejemplo gráfico E2] dibuja un triángulo recto isósceles con su ángulo recto en el punto A y con ambos catetos iguales en longitud al segmento AB.

0153 Cuadrilátero

ID 153 · título recuperado de la página española

Sintaxis

```
quadrilateral(Punto1, Punto2, Punto3, Punto4)
```

Dibuja un cuadrilátero a partir de una serie de cuatro puntos.

Ejemplos

[Ejemplo gráfico E1] dibuja el cuadrilátero ABCD.
[Ejemplo gráfico E2]
[Ejemplo gráfico E3]

0154 Cuadrado

ID 154 · título recuperado de la página española

Sintaxis

```
square(Point1, Point2 [, Var1, Var2])
```

Dibuja un cuadrado a partir de dos vértices consecutivos, Point1 y Point2.

Si se especifican Var1 y Var2, los dos vértices restantes se calculan y se guardan en Var1 y Var2, respectivamente.

Ejemplos

[Ejemplo gráfico E1] dibuja un cuadrado con vértices en (0, 0), (3, 2), (1, 5) y (-2, 3), y guarda los dos últimos vértices en las variables CAS p y q.

0155 Paralelogramo

ID 155 · título recuperado de la página española

Sintaxis

parallelogram(Punto1, Punto2, Punto3)

Dibuja un paralelogramo dado tres de sus vértices. El cuarto punto se calcula automáticamente, pero no se ha definido simbólicamente. Como con la mayoría de los otros comandos de polígono, puede almacenar las coordenadas del cuarto punto en una variable CAS. La orientación del paralelogramo es hacia la izquierda desde el primer punto.

Ejemplos

[Ejemplo gráfico E1] dibuja un paralelogramo cuyos vértices están en (0, 0), (6, 0), (9, 5) y (3,5). Las coordenadas del último punto se calculan automáticamente.

0156 Rombo

ID 156 · título recuperado de la página española

Sintaxis

rhombus(Punto1, Punto2, Ángulo)

Dibuja un rombo, dados dos puntos y un ángulo. Como con muchos de los otros comandos del polígono, puede especificar nombres de variable opcionales CAS para almacenar las coordenadas de los otros dos vértices como puntos.

Ejemplos

[Ejemplo gráfico E1]

[Ejemplo gráfico E2] dibuja un rombo en el segmento AB de forma que el ángulo en el vértice A tiene la misma medida que $\angle DCE$

0157 Rectángulo

ID 157 · título recuperado de la página española

Sintaxis

rectangle(Punto1, Punto2, Punto3)
rectangle(Punto1, Punto2, Realk)

Dibuja un rectángulo dado dos vértices consecutivos y un punto en el lado opuesto del lado definido por los dos primeros vértices o un factor de escala para los lados perpendiculares al primer lado. Como con muchos de los otros comandos del polígono, puede especificar nombres de variable opcionales CAS para almacenar las coordenadas de los otros dos vértices como puntos.

Ejemplos

[Ejemplo gráfico E1]

[Ejemplo gráfico E2] dibuja un rectángulo cuyos dos primeros vértices son los puntos A y B (un lado es el segmento AB). El punto E está en la línea que contiene el lado del rectángulo opuesto al segmento AB.

[Ejemplo gráfico E3] dibuja un rectángulo cuyos dos primeros vértices son los puntos A y B (un lado es el segmento AB). Los lados perpendiculares al segmento AB tienen longitud $3 \cdot AB$. Los puntos tercero y cuarto se almacenan en las variables CAS p y q, respectivamente.

0158 Polígono

ID 158 · título recuperado de la página española

Sintaxis

polygon(Punto1, Punto2, ..., PuntoN)

Dibuja un polígono a partir de una serie de vértices.

Ejemplos

[Ejemplo gráfico E1] dibuja $\triangle ABD$
[Ejemplo gráfico E2]
[Ejemplo gráfico E3]

0159 Polígono regular

ID 159 · título recuperado de la página española

Sintaxis

`isopolygon(Punto1, Punto2, Realn)`

Dibuja un polígono regular a partir de los dos primeros vértices y el número de lados dados, donde el número de lados es mayor a 1. Si el número de lados es 2, entonces se dibuja el segmento.

Puede proporcionar nombres de variables del sistema algebraico computacional para almacenar las coordenadas de los puntos calculados en el orden en que se crearon. La orientación del polígono aparece hacia la izquierda.

Ejemplos

[Ejemplo gráfico E1]
[Ejemplo gráfico E2]
[Ejemplo gráfico E3] dibuja un hexágono regular cuyos dos primeros vértices son los puntos A y B.

0160 Curva

ID 160 · título localizado en HelpEs

Este menú contiene todas las funciones geométricas específicas a las curvas.

0161 Círculo

ID 161 · título recuperado de la página española

Sintaxis

`circle(Punto1, Punto2)`
`circle(Punto1, Punto2-Punto1)`
`circle(equation)`

Dibuja un círculo, dados los extremos del diámetro, el centro y el radio o una ecuación en x e y.

Ejemplos

[Ejemplo gráfico E1] dibuja el círculo con diámetro AB.
[Ejemplo gráfico E2] dibuja el círculo con centro en el punto A y el radio AB.
[Ejemplo gráfico E3] dibuja el círculo de unidad.
[Ejemplo gráfico E5]
[Ejemplo gráfico E6]

Este comando también se puede utilizar para dibujar un arco hacia la derecha.

[Ejemplo gráfico E4] dibuja un cuarto de círculo con diámetro AB.

0162 Circumcircle

ID 162 · título recuperado de la página española

Sintaxis

`circumcircle(Punto1, Punto2, Punto3)`

Dibuja el circuncírculo de un triángulo; es decir, el círculo circunscrito en un triángulo.

Ejemplos

[Ejemplo gráfico E1]
[Ejemplo gráfico E2] dibuja el círculo circunscrito en $\triangle ABC$

0163 Excírculo

ID 163 · título recuperado de la página española

Sintaxis

`excircle(Punto1, Punto2, Punto3)`

Dado tres puntos que definen un triángulo, dibuja el excírculo del triángulo que es tangente al lado definido por los dos últimos puntos y también tangente a las extensiones de los dos lados cuyo vértice común es el primer punto.

Ejemplos

[Ejemplo gráfico E1]

[Ejemplo gráfico E2] dibuja la tangente del círculo al segmento BC y a los rayos AB y AC.

0164 Incírculo

ID 164 · título recuperado de la página española

Sintaxis

`incircle(Punto1, Punto2, Punto3)`

Dibuja el incírculo de un triángulo, la tangente del círculo a los tres lados del triángulo.

Ejemplos

[Ejemplo gráfico E1]

[Ejemplo gráfico E2] dibuja la circunferencia en $\triangle ABC$.

0167 Elipse

ID 167 · título recuperado de la página española

Sintaxis

`ellipse(Equation)`

`ellipse(Point1, Point2, Point3)`

`ellipse(Point1, Point2, k)`

Dibuja una elipse.

`ellipse(Equation)` dibuja la elipse definida por la ecuación dada.

`ellipse(Point1, Point2, Point3)` dibuja la elipse cuyos dos primeros argumentos son los focos. El tercer argumento es un punto de la elipse.

`ellipse(Point1, Point2, k)` dibuja la elipse cuyos dos primeros argumentos son los focos. El tercer argumento es un número real k . Para cualquier punto P de la elipse, la suma de las distancias de P a los dos focos es igual a $2 \cdot k$.

Ejemplos

[Ejemplo gráfico E1] dibuja la elipse definida por la ecuación dada.

[Ejemplo gráfico E2] dibuja la elipse con focos A y B que pasa por el punto C .

[Ejemplo gráfico E3] dibuja la elipse con focos A y B . Para cualquier punto P de la elipse, $AP+BP=6$.

0168 Hipérbola

ID 168 · título recuperado de la página española

Sintaxis

`hyperbola(Punto1, Punto2, Punto3)`

`hyperbola(Punto1, Punto2, Realk)`

Dibuja una hipérbola, dados los focos y un punto sobre la hipérbola o un escalar que es la mitad de la diferencia constante de las distancias desde un punto en la hipérbola para cada uno de los focos.

Ejemplos

[Ejemplo gráfico E1]

[Ejemplo gráfico E2] dibuja la hipérbola cuyos enfoques son los puntos A y B y que pasa a través del punto C .

[Ejemplo gráfico E3] dibuja una hipérbola cuyos enfoques son los puntos A y B . Para cualquier punto P en la hipérbola, $|AP-BP|=6$.

0169 Parábola

ID 169 · título recuperado de la página española

Sintaxis

parabola(Punto, Línea)
parabola(Punto, Realk)
parabola(Expr)

Dibuja una parábola, dado un punto de enfoque y una línea directriz o el vértice de la parábola y un número real que representa la longitud focal

Ejemplos

[Ejemplo gráfico E2] dibuja una parábola cuyo enfoque es el punto A y cuya directriz es la línea B.
[Ejemplo gráfico E3] dibuja una parábola cuyo vértice es el punto A y cuya longitud focal es 1.
[Ejemplo gráfico E1] dibuja el gráfico de la ecuación parabólica $x=y^2-y+2$

0170 Cónica

ID 170 · título recuperado de la página española

Sintaxis

conic(Expr)

Traza la representación gráfica de una sección cónica definida por una expresión en x e y.

Ejemplos

[Ejemplo gráfico E1] dibuja un círculo con centro en (0,0) y radio de 9
[Ejemplo gráfico E2]

0171 Locus

ID 171 · título recuperado de la página española

Sintaxis

locus(Punto, Elemento, [tstep=Valor])

Dados un primer punto y un segundo punto que es un elemento de (un punto en) un objeto geométrico, dibuja el lugar geométrico del primer punto cuando el segundo punto atraviesa su objeto. La declaración de tstep opcional puede utilizarse para controlar el nivel de detalle predeterminado.

0172 Gráfico

ID 172 · título localizado en HelpEs

Este menú contiene un conjunto de comandos para trazar gráficos en la Vista de gráfico de la aplicación de Geometría.

0173 Trazar función

ID 173 · título recuperado de la página española

Sintaxis

plotfunc(Expr)

Utilizado en la aplicación de Geometría en vistas de Gráfico o Simbólicas, o en la vista CAS. Dibuja el gráfico de una función, dada una expresión en la variable independiente x. Tenga en cuenta el uso de la x en minúscula.

Ejemplos

[Ejemplo gráfico E1] dibuja el gráfico de $y=3*\sin(x)$.

0174 Trazar paramétrico

ID 174 · título recuperado de la página española

Sintaxis

plotparam(f(Var)+i*g(Var), Var=Intervalo [, tstep=Valor])

Se utiliza en la Vista simbólica de la aplicación de Geometría. Toma una expresión compleja en una variable y un intervalo para esa variable como argumentos. Interpreta la expresión compleja $f(t)+i*g(t)$ como $x=f(t)$ e $y=g(t)$ y traza la ecuación paramétrica en el intervalo específico en el segundo argumento.

Ejemplos

[Ejemplo gráfico E1] traza el círculo unitario
[Ejemplo gráfico E2] traza un hexágono regular inscrito en el círculo unitario (tenga en cuenta el tstep value)
[Ejemplo gráfico E3]
[Ejemplo gráfico E4]

0175 Trazar Polar

ID 175 · título recuperado de la página española

Sintaxis

```
plotpolar(Expr, Var=Intervalo, [IncrM.])  
plotpolar(Expr, Var, Mín., Máx., [IncrM.])
```

Se utiliza en la aplicación de Geometría para dibujar un gráfico polar en Vista de gráfico.

Ejemplos

[Ejemplo gráfico E1]
[Ejemplo gráfico E2] dibuja la curva polar $r=f(x)$ para x en $[a,b]$

0176 Trazar secuencia

ID 176 · título recuperado de la página española

Sintaxis

```
plotseq(f(Var), Var={Inicio, Xmin, Xmax}, Enteron)
```

Se utiliza en la Vista simbólica de la aplicación de Geometría. Dada una expresión en x y una lista que contiene tres valores, dibuja la línea $y=x$, la representación gráfica de la función definida por la expresión sobre el dominio definido por el intervalo entre los dos últimos valores y dibuja el gráfico de tela de araña para los primeros términos n de la secuencia definida de forma recursiva por la expresión (comenzando en el primer valor).

Ejemplos

[Ejemplo gráfico E1] traza $y=x$ además de $y=1-x/2$ (a partir de $x=-1$ a $x=6$), luego extrae los primeros 5 términos del gráfico telaraña para $u(n)=1-(u(n-1)/2)$, a partir de $u(0)=3$

0177 Trazar implícito

ID 177 · título recuperado de la página española

Sintaxis

```
plotimplicit(Expr, [XIntrvl, YIntrvl])
```

Se utiliza en la aplicación de Geometría, en las Vistas de gráfico o simbólica, o CAS. Traza una curva implícitamente definida desde Expr (en x e y). Específicamente, representa gráficamente $\text{Expr}=0$. Tenga en cuenta el uso de x e y en minúsculas. Con los intervalos opcionales de x e y , representa gráficamente sólo dentro de esos intervalos.

Ejemplos

[Ejemplo gráfico E1]
[Ejemplo gráfico E2] traza un círculo, centrado en el punto $(-5, -4)$, con un radio de 1

0178 Trazar inecuación

ID 178 · título recuperado de la página española

Sintaxis

```
plotinequation(Expr, [x=xrange, y=yrange], [Xstep], [Ystep])
```

Traza el gráfico de la solución de inecuaciones con dos variables.

0179 Trazar campo de direcciones

ID 179 · título recuperado de la página española

Sintaxis

```
plotfield(Expr, VectorVar, [Xstep=Val, Ystep=Val, Opción])
```

Usado en la aplicación Geometría o la Vista CAS. Traza el gráfico del campo de pendiente de la ecuación diferencial $y'=f(x,y)$, donde $f(x,y)$ está contenida en Expr. VectorVar es un vector que contiene las variables. Si VectorVar es de la forma $[x=Interval, y=Interval]$, el campo de pendiente se traza sobre el rango de x y el rango de y especificados. Dados los valores de xstep y de ystep, traza los segmentos del campo de pendiente usando estos incrementos. Si la Opción es 'normalizar', los segmentos del campo de pendiente dibujados tienen la misma longitud.

Ejemplos

[Ejemplo gráfico E1] dibuja el campo de direcciones para $y'=x\sin(y)$, desde -6 a 6 en ambas direcciones, con segmentos que son todos de la misma longitud.

0180 Trazar EDO

ID 180 · título recuperado de la página española

Sintaxis

`plotode(Expr, [Var_1, ..., Var_N], [Val_1, ..., Val_N] [, tstep=Value])`

Utilizado en las vistas Simbólica o Gráfica de la aplicación Geometría o en la vista CAS. Dibuja la solución de la ecuación diferencial $y'=f(\text{Var}_1, \dots, \text{Var}_N)$ que contiene como condición inicial para las variables $\text{Val}_1, \dots, \text{Val}_N$. El primer argumento es la expresión $f(\text{Var}_1, \dots, \text{Var}_N)$, el segundo argumento es el vector de variables y el tercer argumento es el vector de condiciones iniciales. El opcional tstep se puede usar para controlar el nivel de detalle del gráfico.

Ejemplos

[Ejemplo gráfico E1] dibuja el gráfico de la solución a $y'=x\sin(y)$ que pasa por el punto $(-2, 2)$ como su condición inicial.
[Ejemplo gráfico E2]

0181 Trazar lista

ID 181 · título recuperado de la página española

Sintaxis

`plotlist(List)`
`plotlist(Matrix)`

Utilizado en las Vistas de gráfico y simbólica de la aplicación Geometría o CAS, este comando dibuja un conjunto de n puntos y los conecta con segmentos. Los puntos se definen por una matriz $m \times 2$, con las abscisas en la primera fila y las ordenadas en la segunda fila.

Ejemplos

[Ejemplo gráfico E1]
[Ejemplo gráfico E2]
[Ejemplo gráfico E3] dibuja un triángulo

0182 Transformar

ID 182 · título localizado en HelpEs

Este menú contiene todas las funciones geométricas específicas a las transformaciones.

0183 Traslación

ID 183 · título recuperado de la página española

Sintaxis

`translation(Vector, Objeto)`

Traslada un objeto geométrico a lo largo de un vector dado. El vector dado es la diferencia entre dos puntos (módulo).

Ejemplos

[Ejemplo gráfico E1] traslada el objeto A una unidad hacia abajo.
[Ejemplo gráfico E2] traslada el objeto C a lo largo del vector AB.

0184 Reflexión

ID 184 · título recuperado de la página española

Sintaxis

reflection(Línea, Objeto)
reflection(Punto, Objeto)

Refleja un objeto geométrico sobre una línea o a través de un punto. El último se refiere a veces a una media vuelta.

Ejemplos

[Ejemplo gráfico E1] refleja el punto en (1,1) sobre la línea vertical $x=3$ para crear un punto en (5,1).
[Ejemplo gráfico E2] refleja el punto en (3, -2) a través del punto en (1, 1) para crear un punto en (-1, 4).

0185 Rotación

ID 185 · título recuperado de la página española

Sintaxis

rotation(Punto, Ángulo, Objeto)

Gira un objeto geométrico, sobre un punto central determinado, a través de un ángulo determinado.

Ejemplos

[Ejemplo gráfico E1] gira el objeto geométrico etiquetado como K, sobre el punto A, a través de un ángulo igual a $\angle CBD$.

0186 Dilación

ID 186 · título recuperado de la página española

Sintaxis

homothety(Punto, Realk, Objeto)

Amplía un objeto geométrico, con respecto a un punto central, por un factor de escala.

Ejemplos

[Ejemplo gráfico E1] crea una ampliación centrada en el punto A que tiene un factor de escala de 2. Cada punto P en el objeto geométrico B tiene su imagen P' en la raya AP, de tal manera que $AP'=2AP$.
[Ejemplo gráfico E2] crea un punto de imagen en (3,3).

0187 Similitud

ID 187 · título recuperado de la página española

Sintaxis

similarity(Punto, Realk, Ángulo, Objeto)

Amplía y rota un objeto geométrico sobre el mismo punto central.

Ejemplos

[Ejemplo gráfico E1] dilata el punto en (2,0) por un factor de escala de 3 (un punto en (6,0)), luego gira el resultado 90° hacia la izquierda para crear un punto en (0, 6)

0188 Proyección

ID 188 · título recuperado de la página española

Sintaxis

projection(Curva, Punto)

Dibuja la proyección ortogonal de un punto sobre una curva.

Ejemplos

[Ejemplo gráfico E1] crea un punto en el círculo en $(\sqrt{2}, \sqrt{2})$

0189 Inversión

ID 189 · título recuperado de la página española

Sintaxis

`inversion(Punto1, Realk, Punto2)`

Dibuja la inversión de un punto, con respecto a otro punto, por un factor de escala.

Ejemplos

[Ejemplo gráfico E1]

[Ejemplo gráfico E2] dibuja el punto C en la línea AB de modo que $AB \cdot AC = 3$. En este caso, el punto A es el centro de la inversión y el factor de escala es 3. El punto B es el punto cuya inversión se crea.

En general, la inversión del punto A a través del centro C, con el factor de escala k, asigna A a A', de manera que A' esté en la línea CA y $CA \cdot CA' = k$, donde CA y CA' denotan las longitudes de los segmentos correspondientes. Si $k=1$, entonces las longitudes CA y CA' son recíprocas.

0190 Reciprocación

ID 190 · título recuperado de la página española

Sintaxis

`reciprocation(Círculo, [Obj1, Obj2, ...Objn])`

Dados un círculo y un vector de objetos que son líneas o puntos, devuelve un vector donde cada punto es reemplazado con su línea polar y cada línea es reemplazada con su polo, con respecto al círculo.

Ejemplos

[Ejemplo gráfico E1] devuelve `[point(1/2, 1/2) line(y=-x/2+1/2)]`

0191 Cartesiano

ID 191 · título localizado en HelpEs

Este menú contiene comandos que son esencialmente cartesianos en su naturaleza. Estos incluyen las coordenadas de puntos, las ecuaciones de líneas curvas y barras de control deslizante entre otros.

0192 Abscisa

ID 192 · título recuperado de la página española

Sintaxis

`abscissa(P)`
`abscissa(V)`

Devuelve la coordenada x de un punto P o la longitud x de un vector V.

Ejemplos

[Ejemplo gráfico E1]

[Ejemplo gráfico E2]

[Ejemplo gráfico E3] la coordenada x del punto A.

0193 Affix

ID 193 · título recuperado de la página española

Sintaxis

`affix(Point)`
`affix(Vector)`

Devuelve las coordenadas de un punto o las longitudes de x e y de un vector como número complejo.

Ejemplos

[Ejemplo gráfico E1]

[Ejemplo gráfico E2]

Si GA es un punto en (1, -2), entonces `affix(GA)` devuelve $1-2i$.

0955 Coordenadas

ID 955 · título recuperado de la página española

Sintaxis

`coordinates(Punto)`
`coordinates(Vector)`

Dado un punto o un vector de puntos, devuelve una matriz que contiene las coordenadas x e y de esos puntos. Cada fila de la matriz define un punto; la primera columna da las coordenadas x, y la segunda columna contiene las coordenadas y.

0194 Ordenada

ID 194 · título recuperado de la página española

Sintaxis

`ordinate(Punto)`
`ordinate(Vector)`

Devuelve la ordenada x de un punto o la longitud y de un vector.

0195 Coordenadas polares

ID 195 · título recuperado de la página española

Sintaxis

`polar_coordinates(Punto)`

Devuelve un vector que contiene las coordenadas polares de un punto.

0196 Ecuación de

ID 196 · título recuperado de la página española

Sintaxis

`equation(Curva)`
`equation(Punto)`

Devuelve la ecuación cartesiana de una curva en x e y o las coordenadas cartesianas de un punto.

Ejemplos

[Ejemplo gráfico E1]

Si GA es el punto en (0, 0), GB es el punto en (1, 0) y GC se define como un círculo(GA, GB-GA), la ecuación(GC) devuelve $x^2 + y^2 = 1$.

0197 Paramétrica

ID 197 · título recuperado de la página española

Sintaxis

`parameq(Obj)`

Devuelve una ecuación paramétrica para un objeto geométrico. La ecuación paramétrica es válida para todos los números complejos que representan puntos en el objeto.

0198 Medir

ID 198 · título localizado en HelpEs

Este menú contiene todas las funciones específicas de medición geométrica

0203 Ángulo

ID 203 · título recuperado de la página española

Sintaxis

`angle(Vértice, Punto2, Punto3)`

Devuelve la medida de un ángulo dirigido. El primer punto se toma como el vértice del ángulo y los dos puntos siguientes en orden proporcionan la medida y la orientación.

Ejemplos

[Ejemplo gráfico E1] devuelve la medida de $\square$ BAC
[Ejemplo gráfico E2]

0205 Arc Length

ID 205 · título recuperado de la página española

Sintaxis

`arcLen(Expr, Real1, Real2)`

Devuelve la longitud del arco de una curva entre dos puntos de la curva. La curva es una expresión, se declara la variable independiente, y los dos puntos se definen por los valores de la variable independiente.

Este comando también puede aceptar una definición paramétrica de una curva. En este caso, la expresión es una lista de 2 expresiones (la primera para x y la segunda para y) en términos de una tercera variable independiente.

0202 Área

ID 202 · título recuperado de la página española

Sintaxis

`area(Circle)`
`area(Polygon)`
`area(Function, Value1, Value2)`

Devuelve el área de un círculo o polígono. También puede devolver el área bajo una función entre dos valores x.

Ejemplos

En la vista CAS, [Ejemplo gráfico E1] devuelve 64/3.

Si GA se define como el círculo unitario, entonces `area(GA)` devuelve π .

Si GA se define como `plotfunc(4-x2/4)`, entonces `area(GA,-4,4)` devuelve 64/3 o 21.333...

0199 Distancia

ID 199 · título recuperado de la página española

Sintaxis

`distance(Punto1, Punto2)`
`distance(Punto, Curva)`

Devuelve la distancia entre dos puntos o entre un punto y una curva.

Ejemplos

[Ejemplo gráfico E1]

si GA es el punto en (0,0) y GB se define como `plotfunc(4-x2/4)`, entonces la distancia (GA,GB) devuelve $2\sqrt{3}$.

0200 Radio

ID 200 · título recuperado de la página española

Sintaxis

`radius(Círculo)`

Devuelve el radio de un círculo. Para una cónica con centro, devuelve a, b, c y la excentricidad e.

Ejemplos

[Ejemplo gráfico E1]

Si GA es el punto en (0,0), GB es el punto en (1,0) y GC se define como `círculo(GA,GB-GA)`, entonces `radius(GC)` devuelve 1.

0201 Perímetro

ID 201 · título recuperado de la página española

Sintaxis

```
perimeter(Polígono)  
perimeter(Círculo)
```

Devuelve el perímetro de un polígono o la circunferencia de un círculo.

Ejemplos

[Ejemplo gráfico E1]

Si GA es el punto en (0, 0), GB es el punto en (1, 0), y GC se define como circle(GA, GB-GA), entonces perimeter(GC) devuelve 2π .

Si GA es el punto en (0, 0), GB es el punto en (1, 0) y GC se define como square(GA, GB-GA), luego perimeter(GC) devuelve 4.

0204 Slope

ID 204 · título recuperado de la página española

Sintaxis

```
slope(Línea)  
slope(Punto1, Punto2)
```

Dada una línea o dos puntos que determinan una línea, devuelve la pendiente de la línea.

0206 Prueba

ID 206 · título localizado en HelpEs

Este menú contiene todas las funciones específicas de prueba geométrica

0207 Función is_collinear

ID 207 · título recuperado de la página española

Sintaxis

```
is_collinear(Punto1, Punto2, ..., Puntón)
```

Toma un conjunto de puntos como argumento y prueba si son colineales o no. Devuelve 1 si los puntos son colineales; de lo contrario, devuelve 0.

0208 Función is_concyclic

ID 208 · título recuperado de la página española

Sintaxis

```
is_concyclic(Punto1, Punto2, Punto3, Punto4)
```

Toma un conjunto de 4 puntos como argumento y prueba si están todos en el mismo círculo. Devuelve 1 si todos los puntos están en el mismo círculo; de lo contrario, devuelve 0.

0209 Función is_element

ID 209 · título recuperado de la página española

Sintaxis

```
is_element(Punto, Objeto)
```

Prueba si un punto está en un objeto geométrico. Devuelve un número (de 1 a la cantidad de lados) en representación del segmento que contiene el punto y 0 de lo contrario.

0210 Función is_parallel

ID 210 · título recuperado de la página española

Sintaxis

```
is_parallel(Línea1, Línea2)
```

Prueba si dos líneas son paralelas o no. Devuelve 1 si lo son; de lo contrario, devuelve 0.

0211 Función is_perpendicular

ID 211 · título recuperado de la página española

Sintaxis

is_perpendicular(Línea1, Línea2)

Similar a is_orthogonal. Prueba si dos líneas son perpendiculares o no. Devuelve 1 si lo son; de lo contrario, devuelve 0.

0212 Función is_isosceles

ID 212 · título recuperado de la página española

Sintaxis

is_isosceles(Punto1, Punto2, Punto3)

Toma tres puntos y prueba si son vértices o no de un solo triángulo isósceles. Devuelve 0 si no lo son. En caso contrario, devuelve el orden numérico del punto común de los dos lados con la misma longitud (1, 2 o 3). Devuelve 4 si los tres puntos forman un triángulo equilátero.

0213 Función is_equilateral

ID 213 · título recuperado de la página española

Sintaxis

is_equilateral(Punto1, Punto2, Punto3)

Toma tres puntos y prueba si son o no vértices de un único triángulo equilátero. Devuelve 1 si lo son y 0 si no.

0214 Función is_parallelogram

ID 214 · título recuperado de la página española

Sintaxis

is_parallelogram(Punto1, Punto2, Punto3, Punto4)

Prueba si un conjunto de cuatro puntos son vértices de un paralelogramo o no. Devuelve 0 si no lo son. En caso contrario, devuelve 1 si solo forman un paralelogramo, 2 si forman un rombo, 3 si forman un rectángulo y 4 si forman un cuadrado.

0215 Función is_conjugate

ID 215 · título recuperado de la página española

Sintaxis

is_conjugate(Círculo, Punto1, Punto2, [Punto3])
is_conjugate(Línea1, Línea2, Línea3, [Línea4])

Prueba si dos puntos o líneas son conjugados de un círculo dado o no. Devuelve 1 si lo son; de lo contrario, devuelve 0.

0216 Zoom

ID 216 · título localizado en HelpEs

Este menú contiene las opciones de zoom en la Vista de gráfico de la aplicación Geometría.

0217 Funciones simbólicas de geometría

ID 217 · título localizado en HelpEs

Hay dos funciones simbólicas en la aplicación Geometría. Fueron diseñadas para permitir que el usuario edite y elimine definiciones simbólicas de objetos geométricos. Estos comandos son Instruction y DelInstruction.

0218 Instrucción

ID 218 · título recuperado de la página española

El comando Instrucción brinda acceso a la lista de definiciones simbólicas en la Vista simbólica de la aplicación Geometría. Cada definición de la Vista simbólica se numera y 1 es la primera definición.

Instruction(n) devuelve el valor de n instrucción (equivalente a escribir su nombre).

Instruction(n,0) devuelve una representación textual de la instrucción n.

Instruction(n,1) devuelve una lista con toda la información de la instrucción n: {nombre, definición, valor, color, visible(0/1), plotDetail (0-7), completado (0/1), trazado (0/1), leyenda visible (0/1), {} o {frente, dpi, x, y} posición del texto para controles deslizantes y mediciones, datos de animación: {} o {tipo de animación (0-3), incrementos por seg (0-15), pausa (0-15), inicio (real), parar(real), incrementos (real)}

Instruction(n, k) devuelve el elemento k-ésimo - 1 de la lista de metadatos, donde k es entre 2 y 12 inclusive (véase más arriba).

Instruction(n):= establece el valor de la instrucción n de una cadena.

Instruction(n,1):= establece todos los datos de las instrucciones mediante el mismo formato de lista según se describe antes.

Instruction(n, k):= establece objeto de metadatos k- 1 a un valor específico, donde k se encuentra entre 2 y 12 inclusive (véase más arriba).

Si n=0 o si n es mayor que el número de instrucciones actuales, agrega una instrucción al final de la lista de instrucciones.

Si n<0, inserta una instrucción en la posición -n.

n también puede ser el nombre de la variable, ya sea que use 'nombre' o "nombre"

0219 Eliminar Instrucción

ID 219 · título recuperado de la página española

Sintaxis

DelInstruction([a [, b]])

El comando DelInstruction permite eliminar una o más definiciones simbólicas de objetos geométricos en la vista simbólica de la aplicación de Geometría. Cada definición en la vista simbólica está numerada, siendo 1 la primera definición.

DelInstruction() borra TODAS las instrucciones.

DelInstruction(a) borra la instrucción a.

DelInstruction(a,b) borra las instrucciones de a a b.

0220 Aplicación Hoja de cálculo

ID 220 · título localizado en HelpEs

La aplicación Hoja de cálculo ofrece una cuadrícula de celdas para que introduzca contenido (números, texto, expresiones, etc.) y para realizar ciertas operaciones con dicho contenido.

Pulse [[Inicio]] o presione Enter para iniciar la aplicación.

La aplicación se abre en la Vista numérica. Las teclas de menú son:

- [[Formato]] abre el menú Formato
- [[Ir a]] pasa a una celda específica
- [[Selección]] activa o desactiva el modo de selección
- [[Ir]] determina adónde va la selección después de presionar Enter. Alterna entre las opciones a la derecha y hacia abajo.

Los siguientes gestos son compatibles en la Vista numérica:

- toque para seleccionar una celda
- mantener pulsado y arrastre para seleccionar una matriz rectangular de celdas
- arrastre para desplazar la ventana
- junte los dedos para iniciar el desplazamiento cinético de la ventana en la dirección deseada
- abra/cierre los dedos de forma vertical para aumentar la altura de la fila que contiene la celda seleccionada actualmente
- abra/cierre los dedos de forma horizontal para aumentar / disminuir el ancho de la columna que contiene la celda seleccionada actualmente

0221 Navegación y selección

ID 221 · título localizado en HelpEs

Puede moverse por una hoja de cálculo utilizando la rueda, pulsando la pantalla o arrastrando el dedo por esta, o pulsando [[Ir a]] y especificando la celda hasta la que quiere desplazarse.

Seleccione una celda desplazándose hasta ella. También puede seleccionar una columna entera si toca la letra de la columna y una fila entera si toca el número de la fila. También puede seleccionar toda la hoja de cálculo: toque la celda no numerada en la esquina superior izquierda de la hoja de cálculo (tiene el logotipo de HP).

Puede seleccionar un bloque de celdas si presiona la celda que será la de la esquina de la selección y, tras un segundo, arrastra el dedo a la celda opuesta diagonalmente. También puede seleccionar un bloque de celdas si se desplaza a una celda de la esquina, pulsa [[Seleccionar]] y utiliza la rueda para desplazarse hasta la celda opuesta diagonalmente. Si pulsa [[Selec.]] u otra celda, anulará la selección.

0222 Referencia de las celdas

ID 222 · título localizado en HelpEs

Puede referirse al valor de una celda en fórmulas como si se tratara de una variable. La referencia de una celda consiste en sus coordenadas de columnas y filas. Las referencias pueden ser absolutas o relativas. Una referencia absoluta se escribe como \$C\$R (donde C es la letra de columna y R el número de fila). Por lo tanto, \$B\$7 es una referencia absoluta. En una fórmula, siempre hará referencia a los datos de la celda B7 sin importar dónde se sitúe la fórmula (o su copia). Por otra parte, B7 es una referencia relativa. Se basa en la posición relativa de las celdas. Por lo tanto, una fórmula en la que, por ejemplo, B8 hace referencia a B7, si la fórmula se copia a C8, la fórmula hará referencia a C7 en vez de a B7.

También se pueden especificar rangos de celdas, como en C6:E12, y también columnas enteras (E:E) o filas enteras (\$3:\$5). Tenga en cuenta que el componente alfabético de los nombres de las columnas pueden estar en mayúsculas o en minúsculas, excepto para las columnas g, l, m y z. Estas deben estar en minúsculas. Por lo tanto, puede hacerse referencia a la celda B1 como B1, b1, \$B\$1 o \$b\$1, mientras que a la celda M1 solo puede hacerse referencia como m1 o \$m\$1. (G, L, M y Z son nombres reservados para objetos de gráficos, listas, matrices y números complejos).

Fila, Col y Celda también son variables que se pueden usar para referencias. Vea las Variables de hoja de cálculo para obtener más información.

0223 Denominación de las celdas

ID 223 · título localizado en HelpEs

Las celdas, filas y columnas pueden recibir un nombre. El nombre se puede utilizar a continuación en una fórmula. Una celda con nombre se muestra con un borde azul.

Para dar un nombre a una celda, fila o columna:

1. Seleccione la celda, fila o columna.

Método 1:

2. Introduzca un nombre
3. Toque [[Nombre]] en el menú

Método 2:

2. Toque [[Form.]] y seleccione [[Nombre]].
3. Introduzca un nombre y toque [[OK]].

Uso de los nombres en los cálculos

El nombre que le da a una celda, fila o columna se puede utilizar en una fórmula. Por ejemplo, si nombra una celda TOTAL, podría introducir en otra celda la fórmula =TOTAL*11.

0224 Introducción de contenido

ID 224 · título localizado en HelpEs

Sintaxis

Row
Cell

Una celda puede contener un objeto de la calculadora válido: un número real (3.14), un número complejo ($a + bi$), un entero (#1Ah), una lista ({1, 2}), una matriz o un vector ([1, 2]), una cadena ("texto"), una unidad (2_m) o una expresión (es decir, una fórmula).

Desplácese hasta la celda en la que desea configurar contenido para empezar a introducir el contenido tal y como haría en la vista de Inicio. Presione Enter cuando haya terminado. También puede introducir contenido en varias celdas con una sola entrada. Seleccione las celdas, introduzca el contenido (por ejemplo: =Row*3) y pulse Enter.

Lo que introduzca en la línea de entrada se evaluará en el momento que pulse Enter, y el resultado aparecerá en la celda o celdas. Sin embargo, si quiere conservar la fórmula secundaria, precédala con = (Shift de la tecla de separador decimal). Por ejemplo, imagine que desea agregar la celda A1 (que contiene el número 7) a la celda B2 (que contiene el número 12). Si introduce A1+B2 en la celda A4, por ejemplo, el resultado dará 19, igual que si introduce =A1+B2 en la celda A5. Sin embargo, si el valor de A1 (o B2) cambia, el valor de A5 también, pero no el valor de A4. Esto se debe a que la expresión (o fórmula) se conservaba en A5.

Para ver si una celda contiene solo el valor que se muestra en ella o si hay una fórmula secundaria que genera el valor, desplace el cursor hasta la celda. La línea de entrada mostrará una fórmula en el caso de que la haya.

Al introducir una fórmula utilizando =, tiene la opción de que se evalúe el resultado numéricamente o utilizar el sistema algebraico computacional para generar resultados exactos o simbólicos. Cuando haya introducido =, observará que aparece un botón de menú [[CAS]]. Si activa este botón tocándolo (un punto indicará que está activo), los resultados se evaluarán mediante el sistema algebraico computacional al pulsar el botón Enter. Al seleccionar una celda, fila o columna que tiene una fórmula del sistema algebraico computacional activo, aparecerá "CAS" en letras rojas en la parte superior del signo igual.

Se puede utilizar una sola fórmula para generar un valor para cada celda en una columna o fila. Por ejemplo, desplácese a C (la celda que es el encabezado de la columna C), introduzca =SIN(Row) y pulse Enter. Cada celda de la columna mostrará el seno del número de la fila de la celda. Un proceso muy similar permite rellenar cada celda de una fila con la misma fórmula. Tenga en cuenta que no se ha introducido contenido en estas celdas.

Por ejemplo, configure A1 en 1 y A en =Cell(Row-1,1)+Cell(Row-2,1) para mostrar una secuencia Fibonacci en la columna A.

También puede agregar una fórmula una vez y aplicarla a cada celda de la hoja de cálculo. Para ello, coloque la fórmula en la celda superior izquierda (la celda con el logotipo de HP en su interior).

Por ejemplo: =COMB(Row, Cell) creará una hoja con el triángulo de Pascal

0225 Importación desde la aplicación Estadística

ID 225 · título localizado en HelpEs

Puede importar datos de las aplicaciones Var 1 estadística y Var 2 estadística (y desde cualquier aplicación personalizada de una aplicación estadística). El siguiente procedimiento importa el conjunto de datos D1 de la aplicación Var 1 estadística.

1. Seleccione una celda
2. Introduzca Statistics_1Var.D1
3. Presione Enter

La columna se completa con los datos de la aplicación estadística, empezando con la celda seleccionada en el paso 1. Cualquier dato que esté en esa columna se sustituirá por los datos importados.

Por otro lado, si empieza la fórmula del Paso 2 con =, la lista entera D1 de la aplicación Var 2 estadística se pegará en la celda seleccionada.

0226 Ordenar celdas

ID 226 · título localizado en HelpEs

Para ordenar un grupo de celdas:

1. Seleccione las celdas que desea ordenar.
2. Toque [[Orden.]]; aparece un menú que permite seleccionar la columna por la que desea realizar la ordenación.
3. Seleccione la columna. Aparece un submenú que permite seleccionar entre dos opciones de ordenación: ascendente (↑) y descendente (↓).
4. Seleccione una opción de ordenación. Los valores en las celdas seleccionadas se ordenan en consonancia.

0227 Copiar y pegar

ID 227 · título localizado en HelpEs

1. Para copiar una o más celdas, selecciónelas y presione Shift View (Copy).
2. Muévase a la ubicación deseada y presione Shift Menu (Paste).
3. Se abre el cuadro de elección Pegar y sus elecciones aparecen seleccionadas y en forma de lista. Pulse sobre la selección (o presione la rueda hacia la derecha) para ver una lista de opciones a pegar. Puede elegir pegar el valor, la fórmula, el formato, el valor y el formato, o bien la fórmula y el formato.
4. Pulse la opción a pegar o utilice la rueda para seleccionar una opción y presione Enter (o pulse [[OK]]). Se ha completado la operación de pegado.

0228 Referencias externas

ID 228 · título localizado en HelpEs

Puede referirse a los datos de una hoja de cálculo desde fuera de la aplicación Hoja de cálculo utilizando la columna y la fila de referencia. Por ejemplo, en la vista de Inicio puede referirse a la celda A6 en la hoja de cálculo integrada introduciendo A6. Por lo tanto, la fórmula 6*A6 multiplicará el valor que haya actualmente en la celda A6 por 6.

Si la hoja de cálculo no es la aplicación actual, igualmente puede usar el contenido de la Celda al calificar por completo el nombre de la celda mediante AppName.CellName. Por ejemplo, Hojadecálculo.A6 o Ahorros.A6 si ha guardado su hoja de cálculo como 'Ahorros'.

También se puede realizar una referencia externa a una celda con nombre, como en 5*Savings.TOTAL. De la misma forma, también puede introducir referencias a las celdas de la hoja de cálculo en el sistema algebraico computacional.

Tenga en cuenta que una referencia a un nombre de la hoja de cálculo distingue entre mayúsculas y minúsculas.

0229 Expresiones en hoja de cálculo

ID 229 · título localizado en HelpEs

Cualquier expresión válida en el inicio del sistema algebraico computacional se puede utilizar en la aplicación Hoja de cálculo. Esto incluye el uso de variables de inicio, sistema algebraico computacional, usuario y aplicación.

0230 Cómo dar un nombre a una celda, fila o columna

ID 230 · título localizado en HelpEs

Introduzca el nombre de la celda, fila o columna, y toque [[OK]].

0231 Ir a

ID 231 · título localizado en HelpEs

Puede ir directamente a cualquier celda de la hoja de cálculo tocando [[Ir a]]. Introduzca la definición de la celda a la que desea ir y toque [[OK]].

La definición puede ser la referencia a una celda (como B7) o el nombre determinado de una celda.

La definición también puede ser una selección del tipo A1:F7, en cuyo caso, todas las celdas se seleccionan.

1519 CHOOSE (hoja de cálculo)

ID 1519 · título localizado en HelpEs

Sintaxis

=CHOOSE(Variable_o_número, "título", lista)

La función CHOOSE cuenta con una aplicación especial en la hoja de cálculo.

Al colocar una función CHOOSE como fórmula de la celda sucederán 4 cosas:

- El valor de la celda (resultado de la evaluación de CHOOSE) será el elemento seleccionado en el cuadro de selección
- La celda mostrará un menú desplegable
- Al tocar en la celda se abrirá la opción y se le permitirá cambiar la selección
- Cuando el usuario cambia el elemento seleccionado en la opción en toda la IU, se llamará la función title_Changed del programa de la aplicación.

Observe que la 'lista' puede ser permanente {"up", "down"}, calculada $\text{MAKELIST}(X^2, X, 1, 10)$ o extraída de la hoja de cálculo (A1:A5).

El primer parámetro de la opción puede ser un nombre de variable o una referencia de celda.

Ejemplos

Si la fórmula en B1 es

[Ejemplo gráfico E1]

Crea un cuadro de opción que permite que el usuario elija Arriba o Abajo

Si la fórmula en B2 es

[Ejemplo gráfico E2]

Crea un cuadro de opción similar al ejemplo anterior, excepto que el elemento seleccionado se almacena en C2

Si la fórmula en B3 es

[Ejemplo gráfico E3]

Permite que el usuario elija entre los primeros 10 enteros cuadrados

Y, finalmente, asume que la hoja de cálculo cuenta con datos en las celdas A1 a A5

Si la fórmula en B4 es

[Ejemplo gráfico E4]

Permite que el usuario elija entre las primeras 5 celdas de la Columna A

0232 Opciones de formato

ID 232 · título localizado en HelpEs

Las opciones de formato aparecen cuando toca `[[Form.]]` y se aplican a todos los elementos seleccionados en ese momento: una celda, un bloque, una columna, una fila o la hoja de cálculo completa.

Las opciones dependerán de los elementos seleccionados. La lista completa de opciones es la siguiente:

- Nombre: muestra un formulario de entrada para que introduzca un nombre para la selección
- Formato de los números: Automático, Estándar, Fijo, Científico o Ingeniería
- Tam. fuente: Automático o de 10 a 22 puntos
- Color: color del contenido (texto, número, etc.) en las celdas seleccionadas; la opción con puntos grises representa la opción Automático
- Rellenar: color de fondo que rellena las celdas seleccionadas; la opción con puntos grises representa la opción Automático
- Alinear ⇄: alineación horizontal; Automático, Izquierda, Centro, Derecha
- Alinear †: alineación vertical; Automático, Arriba, Centro, Fondo
- Columna ⇄: muestra un formulario de entrada para especificar el ancho necesario de las columnas seleccionadas; solo está disponible si ha seleccionado toda la hoja de cálculo o una o más columnas enteras
- Fila †: muestra un formulario de entrada para que usted especifique la altura requerida de las filas seleccionadas; solo está disponible si ha seleccionado toda la hoja de cálculo o una o más de las filas enteras.
- mostrar “ ”: muestra comillas alrededor de las cadenas en el cuerpo de la hoja; Automático, Sí, No
- Libro de texto: muestra las fórmulas en formato de libro de texto; Automático, Sí, No
- Almacenamiento: active esta opción para acelerar los cálculos de las hojas de cálculo con muchas fórmulas; solo disponible si ha seleccionado la hoja de cálculo entera

0235 Variables de hoja de cálculo

ID 235 · título localizado en HelpEs

Aparte de las variables de modos (que son comunes a todas las aplicaciones), la aplicación Hoja de cálculo tiene las siguientes variables numéricas:

- ColWidth
- RowHeight
- Row
- Col
- Cell

0236 Variable ColWidth

ColWidth(Integer) le permite establecer y obtener el ancho de las columnas.

Integer1 ► ColWidth(Integer2) establece el ancho de la columna Integer2 (A=1, B=2, etc.) para los píxeles de Integer1. Aquí, tanto Integer1 como Integer2 son positivos.

Si no se especifica Integer2, establece el ancho predeterminado para las columnas en la hoja de cálculo en los píxeles de Integer1.

ColWidth(Integer) devuelve el ancho de la columna especificado por Integer (A=1, B=2, etc.).

También puede establecer el ancho de columna en el menú [[Formato]].

0237 Variable RowHeight

RowHeight(Integer) le permite establecer y obtener la altura de las filas.

Integer1 ► RowHeight(Integer2) establece la altura de los píxeles de la fila Integer2 para Integer1.

Si no se especifica Integer2, establece la altura predeterminada para las filas de la hoja de cálculos de los píxeles de Integer1.

RowHeight(Integer) devuelve la altura de la fila especificada por Integer.

También puede establecer el ancho de columna en el menú [[Formato]].

0238 Variable Row

Row es una variable que indica el número de la fila de la celda actualmente calculado.

Esto se utiliza principalmente cuando se crea la expresión genérica que necesita para trabajar en la hoja de cálculo, para las columnas o expresiones completas de la hoja de cálculo.

Pasos de ejemplo:

1. Seleccione la columna A
2. Escriba =expand((x+1)^Row)
3. Pulse el botón de menú [[CAS]]
4. Pulse el botón de menú [[OK]]

La columna A contendrá ahora las expansiones de $(x+1)^1$, $(x+1)^2$, $(x+1)^3$, etc.

0239 Variable Col

Col es una variable que indica el número de columna de la celda calculada actualmente (A=1, B=2, etc.).

Esto se utiliza principalmente cuando se crea una expresión genérica que necesita funcionar en cualquier lugar de la hoja de cálculo, o para las filas o expresiones completas de hoja de cálculo.

Pasos de ejemplo:

1. Toque en la esquina superior izquierda de la hoja de cálculo (dónde está el logotipo de HP) para seleccionar la hoja de cálculo entera
2. Tipo =COMB(Row-1,Col-1)
3. Pulse el botón del menú [[OK]]

Ahora, la hoja de cálculo se llenará con el triángulo de Pascal.

0240 Referencias de celda y celda

Sintaxis

Cell(Row, Col [, n])

Hace referencia a una celda, fila, columna, selección o a la propia hoja de cálculo y permite acceder a su valor, fórmula y otros atributos.

En la mayoría de los casos, las celdas pueden referenciarse directamente mediante sus nombres estándar, como A1 o D6. La forma Cell(Row, Col) resulta útil en fórmulas más avanzadas, especialmente cuando los números de fila o columna se calculan dinámicamente.

Parámetros

- Row: Número de fila. Un valor de 0 hace referencia a toda la columna o, si Col también es 0, a la propia hoja de cálculo.
- Col: Número de columna, donde A = 1, B = 2, etc. Un valor de 0 hace referencia a toda la fila o, si Row también es 0, a la propia hoja de cálculo.
- n: Número de atributo opcional. Si se omite, se accede al valor o a la fórmula asociados con la referencia.

Referencias básicas de celda

Cell(Row, Col) equivale a una referencia de celda estándar. Por ejemplo, Cell(1, 1) equivale a A1.

A1:=100 almacena el valor 100 en A1.

A1:=A2+A3 almacena el resultado actual de A2+A3 en A1.

A1:='A2+A3' establece la fórmula de A1 en A2+A3.

Niveles de referencia

Un número de fila o columna igual a 0 permite acceder a un nivel más amplio de la hoja de cálculo:

- Cell(Row, Col): Una sola celda.
- Cell(0, Col): La columna especificada.
- Cell(Row, 0): La fila especificada.
- Cell(0, 0): La propia hoja de cálculo.

Los atributos se pueden leer o modificar en cualquiera de estos niveles.

Lectura de valores y atributos

Si se omite n y la referencia no se utiliza como destino de almacenamiento, se devuelve el valor asociado con la referencia.

Para una sola celda, se devuelve su valor.

Para una fila o columna, se devuelve una lista de valores.

Para una selección rectangular, se devuelve una lista de listas, una lista por cada columna.

Se considera que una celda sin ningún valor asociado tiene un valor de 0.

Si se especifica n, se devuelve el atributo correspondiente. Por ejemplo:

Cell(1, 1, 8) devuelve la alineación horizontal definida para A1.

Cell(0, 1, 8) devuelve la alineación horizontal definida para la columna A.

Cell(1, 0, 8) devuelve la alineación horizontal definida para la fila 1.

Cell(0, 0, 8) devuelve la alineación horizontal definida para toda la hoja de cálculo.

Modificación de valores y atributos

Si se omite n y la referencia se utiliza como destino de almacenamiento, se modifica la expresión asociada con el destino.

Si se asigna una única expresión de origen a varios destinos, la expresión se duplica para todos los destinos y las referencias relativas de celda se ajustan según sea necesario.

Por ejemplo:

A1:=1;

A2:A10:='A1+1';

Si se especifica n, se modifica el atributo correspondiente. Por ejemplo:

Cell(0, 0, 8):=1 establece la alineación horizontal de la hoja de cálculo en Centro.

Cell(0, 1, 8):=2 establece la alineación horizontal de la columna A en Derecha.

Cell(1, 0, 8):=0 establece la alineación horizontal de la fila 1 en Izquierda.

Cell(1, 1, 8):=1 establece la alineación horizontal de A1 en Centro.

Establecer un atributo en -1 elimina la configuración explícita de ese nivel y hace que el valor se herede.

Atributos de celda

Cuando se especifica n, se puede acceder a los siguientes atributos de celda:

- -1: Devuelve los atributos 0 a 10 como una lista. Si no hay nada definido, devuelve -1.
- 0: Valor. Solo lectura; el valor calculado no se puede establecer directamente.
- 1: Fórmula.
- 2: Nombre.
- 3: Formato de número: Estándar = 0, Fijo = 1, Científico = 2, Ingeniería = 3, Flotante = 4, Redondeado = 5, no especificado = -1.

- 4: Número de lugares decimales: 1 a 11, no especificado = -1.
- 5: Fuente: 0 a 6, no especificado = -1.
- 6: Color de primer plano, no especificado = -1.
- 7: Color de fondo, no especificado = -1.
- 8: Alineación horizontal: Izquierda = 0, Centro = 1, Derecha = 2, no especificado = -1.
- 9: Alineación vertical: Arriba = 0, Centro = 1, Abajo = 2, no especificado = -1.
- 10: Mostrar cadenas entre comillas: Sí = 0, No = 1, no especificado = -1.
- 11: Modo Libro de texto, en oposición al modo algebraico: Sí = 0, No = 1, no especificado = -1.

Fórmula y valor a nivel de hoja de cálculo

También se puede asociar una fórmula con la propia hoja de cálculo mediante el atributo 1.

Cell(0, 0, 1):='2+3'

establece la fórmula de la hoja de cálculo en 2+3. Las celdas que no tienen su propia definición utilizan la fórmula de la hoja de cálculo y, por tanto, muestran su resultado.

El valor calculado de la fórmula de la hoja de cálculo se puede leer mediante:

Cell(0, 0, 0)

Herencia de atributos

Los atributos se pueden definir de forma independiente para una celda, columna, fila o para toda la hoja de cálculo. Cuando el mismo atributo está definido en más de un nivel, tiene prioridad la definición más específica:

celda > columna > fila > hoja de cálculo

Un valor de atributo de -1 significa que no está especificado. En este caso, el atributo se hereda del siguiente nivel aplicable. Si no se define ningún valor explícito a nivel de hoja de cálculo, se utiliza el valor predeterminado del sistema.

Sintaxis de referencias

Las referencias pueden utilizar la sintaxis RC o nombres asignados a celdas, filas o columnas.

Las referencias RC válidas incluyen:

- [R][C](n)
- [R1:C1:[R2][C2](n)]
- [R:[R](n)] o RowName[(n)]
- [C:[C](n)] o ColName[(n)]

R, R1 y R2 representan nombres o números de fila. C, C1 y C2 representan nombres o números de columna.

Las celdas y selecciones también pueden referenciarse mediante CellName. Para obtener más información sobre las referencias de celda con nombre y su sintaxis, consulte CellName.

Ejemplos

Cell(0, 0, 8):=0;

Cell(0, 1, 8):=2;

Cell(1, 1, 8):=1;

La hoja de cálculo está alineada a la izquierda, la columna A está alineada a la derecha y A1 está centrada.

Cell(1, 1, 8):=-1

elimina la alineación explícita de A1, haciendo que herede la alineación a la derecha de la columna A.

0241 Variable Cursor

ID 241 · título recuperado de la página española

Sintaxis

Cursor
Cursor(n)

Cursor devuelve una lista que representa la posición del cursor y la ubicación donde se detiene la selección.

Cursor(n) devuelve el n-avo objeto de la lista que Cursor habría devuelto

Cursor:= {fila, col, [selectionRow, selectionCol]}

Establece la posición del cursor

Si se especifica una selección al establecer Cursor, la selección se activa o se desactiva

0242 Variable CellName

ID 242 · título recuperado de la página española

Sintaxis

```
CellName  
CellName("nombre")
```

CellName devuelve la lista de todas las celdas mencionadas en la hoja de cálculo

CellName("name") devuelve una lista con la fila y columna de la celda con nombre, si existe. De lo contrario, devuelve 0.

0243 Funciones de la aplicación Hoja de cálculo

ID 243 · título localizado en HelpEs

La hoja de cálculo dispone de un conjunto de funciones de aplicación específicas.

Estas funciones se pueden organizar en 2 grupos.

1. Funciones que se centran en celdas vacías (pero que siguen devolviendo 0, por ejemplo, AVERAGE)
2. Funciones diseñadas para agilizar los cálculos devolviendo más de 1 resultado al mismo tiempo.

0244 Función SUM

ID 244 · título recuperado de la página española

Sintaxis

```
SUM([Entrada])
```

Calcula la suma de un rango de números.

Por ejemplo, SUM(B7:B23) devuelve la suma de los números en el rango B7 a B23. También puede especificar un bloque de celdas, como en SUM(B7:C23).

Si una celda en el rango especificado contiene un objeto no numérico, devuelve un error.

0245 Función AVERAGE

ID 245 · título recuperado de la página española

Sintaxis

```
AVERAGE([Entrada])
```

Calcula el promedio aritmético de un rango de números.

Por ejemplo, AVERAGE(B7:B23) devuelve la media aritmética de los números en el rango B7 a B23. También puede especificar un bloque de celdas, como en AVERAGE(B7:C23).

Se devuelve un error si una celda en el rango especificado contiene un objeto no numérico. Se omiten las celdas vacías.

0246 Función AMORT

ID 246 · título recuperado de la página española

Sintaxis

```
AMORT(range, n, i, pv, pmt [, grouping, ppyr, cpyr, beg, fix] [, configuration])
```

Crea una tabla de amortización para un préstamo o una inversión según las condiciones de pago especificadas.

Parámetros

- range: el rango de celdas donde se colocan los resultados. Si se especifica una sola celda, el rango se calcula automáticamente.
- n: el número de períodos de pago.
- i: la tasa de interés.
- pv: el valor actual.

- **pmt:** el pago por período.
- **grouping:** el número de períodos de pago agrupados en cada fila de la tabla de amortización. Por ejemplo, **grouping=1** crea una fila por período de pago, y **grouping=3** crea una fila por cada 3 períodos de pago. El valor predeterminado es **ppyr**.
- **ppyr:** el número de pagos por año. El valor predeterminado es 12.
- **cpyr:** el número de períodos de capitalización por año. El valor predeterminado es **ppyr**.
- **beg:** especifica cuándo se realiza el pago. Use 1 cuando el pago se realiza al comienzo de cada período; de lo contrario, use 0. El valor predeterminado es 0.
- **fix:** el número de decimales que se muestran en los resultados de los cálculos. De forma predeterminada, se determina según la configuración actual.
- **configuration:** una cadena que especifica si se crea una fila de encabezados de columna y qué columnas de resultados se incluyen. Los códigos de configuración se escriben juntos en una sola cadena. H: crea una fila de encabezados de columna. h: incluye la columna de etiquetas de fila. S: incluye el período inicial de cada fila. E: incluye el período final de cada fila. P: incluye el capital para el período o grupo de períodos. B: incluye el saldo al final del período o grupo de períodos. I: incluye los intereses para el período o grupo de períodos.

Ejemplo de configuración:

"HhEP" crea una fila de encabezados de columna e incluye las columnas de etiqueta de fila, Final y Capital.

Ejemplos

[Ejemplo gráfico E1] Crea una tabla de amortización utilizando los valores opcionales predeterminados.

[Ejemplo gráfico E2] Crea una tabla de amortización utilizando los valores opcionales predeterminados, con una fila de encabezados de columna y las columnas de etiqueta de fila, Final y Capital.

[Ejemplo gráfico E3] Crea una tabla de amortización para 36 períodos de pago, agrupados en filas de 6 períodos, con 12 pagos por año, 12 períodos de capitalización por año, pagos al final del período, 2 decimales y las columnas Inicial, Final, Capital, Intereses y Saldo.

0247 Función STAT1

ID 247 · título recuperado de la página española

Sintaxis

STAT1(input_range [, mode] [, configuration])

Proporciona estadísticas de una variable para uno o varios conjuntos de datos.

Parámetros

- **input_range:** el rango de celdas que contiene los datos de origen, por ejemplo, A1:D8. Si se especifica más de una columna, cada columna se trata como un conjunto de datos independiente. Si solo se selecciona una fila, se trata como un único conjunto de datos.
- **mode:** especifica cómo se interpretan los datos de entrada. Si se seleccionan dos columnas y no se especifica mode, el valor predeterminado es 2. 1: trata cada columna como un conjunto de datos independiente. 2: trata las columnas por pares como datos de frecuencia. En cada par, la segunda columna especifica la frecuencia de los valores de la primera columna.
- **configuration:** una cadena que especifica qué valores estadísticos se incluyen, el orden en que aparecen y si se crean encabezados de fila o de columna. Introduzca los códigos separados por espacios y en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. H: crea encabezados de columna. h: crea encabezados de fila. MeanX: incluye la media aritmética. Σ : incluye la suma. Σ^2 : incluye la suma de los cuadrados. s: incluye la desviación estándar de la muestra. s^2 : incluye la varianza de la muestra. σ : incluye la desviación estándar de la población. σ^2 : incluye la varianza de la población. serr: incluye el error estándar. ss: incluye la suma de las desviaciones al cuadrado. n: incluye el número de valores de datos. min: incluye el valor mínimo. q1: incluye el primer cuartil. med: incluye la mediana. q3: incluye el tercer cuartil. max: incluye el valor máximo.

Ejemplo de configuración:

"h n σ " crea encabezados de fila, coloca el número de valores de datos en la primera fila de resultados y la desviación estándar de la población en la segunda fila de resultados.

0248 Función STAT2

ID 248 · título recuperado de la página española

Sintaxis

STAT2(input_range [, mode] [, configuration])

Proporciona estadísticas de dos variables para uno o varios conjuntos de datos emparejados.

Parámetros

- **input_range**: el rango de celdas que contiene los datos de origen, por ejemplo, A1:D8. En el modo 1, cada par de columnas se trata como un conjunto de datos emparejado independiente. En el modo 2, cada grupo de tres columnas se trata como un conjunto de datos de frecuencia independiente. Si solo se selecciona un par de columnas o un grupo de tres columnas, se trata como un único conjunto de datos.
- **mode**: especifica cómo se interpretan los datos de entrada. Si se seleccionan tres columnas y no se especifica mode, el valor predeterminado es 2. 1: trata cada par de columnas como un conjunto de datos emparejado que contiene valores x e y. 2: trata cada grupo de tres columnas como datos de frecuencia. Las dos primeras columnas contienen los valores x e y emparejados, y la tercera columna especifica su frecuencia.
- **configuration**: una cadena que especifica qué valores estadísticos se incluyen, el orden en que aparecen y si se crean encabezados de fila o de columna. Introduzca los códigos separados por espacios y en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. H: crea encabezados de columna. h: crea encabezados de fila. MeanX: incluye la media aritmética de los valores x. Σx : incluye la suma de los valores x. Σx^2 : incluye la suma de los cuadrados de los valores x. sx: incluye la desviación estándar de la muestra de los valores x. sx^2 : incluye la varianza de la muestra de los valores x. σx : incluye la desviación estándar de la población de los valores x. σx^2 : incluye la varianza de la población de los valores x. serrx: incluye el error estándar de los valores x. ssx: incluye la suma de las desviaciones al cuadrado de los valores x. n: incluye el número de pares de valores de datos. $\bar{y}$: incluye la media aritmética de los valores y. Σy : incluye la suma de los valores y. Σy^2 : incluye la suma de los cuadrados de los valores y. sy: incluye la desviación estándar de la muestra de los valores y. sy^2 : incluye la varianza de la muestra de los valores y. σy : incluye la desviación estándar de la población de los valores y. σy^2 : incluye la varianza de la población de los valores y. serry: incluye el error estándar de los valores y. ssy: incluye la suma de las desviaciones al cuadrado de los valores y. Σxy : incluye la suma de los productos de los valores x e y emparejados.

Ejemplo de configuración:

"h n MeanX σy " crea encabezados de fila, coloca el número de pares de valores de datos en la primera fila de resultados, la media aritmética de los valores x en la segunda fila de resultados y la desviación estándar de la población de los valores y en la tercera fila de resultados.

0249 Función REGRS

ID 249 · título recuperado de la página española

Sintaxis

REGRS(input_range [, model] [, configuration])

Ajusta uno o varios conjuntos de datos de entrada al modelo de regresión especificado por model.

Parámetros

- **input_range**: el rango de celdas que contiene los datos de origen, por ejemplo, A1:D8. El rango debe contener un número par de columnas. Cada par de columnas se trata como un conjunto independiente de puntos de datos.
- **model**: especifica el modelo de regresión utilizado para ajustar los datos de entrada. El valor predeterminado es 1. 1: modelo lineal: $y=sl*x+int$. 2: modelo logarítmico: $y=sl*\ln(x)+int$. 3: modelo exponencial: $y=int*\exp(sl*x)$. 4: modelo potencial: $y=int*x^{sl}$. 5: modelo exponencial de base variable: $y=int*sl^x$. 6: modelo recíproco: $y=sl/x+int$. 7: modelo logístico: $y=L/(1+a*\exp(b*x))$. 8: modelo sinusoidal: $y=a*\sin(b*x+c)+d$. 9: modelo cuadrático: $y=cx^2+bx+a$. 10: modelo cúbico: $y=dx^3+cx^2+bx+a$. 11: modelo polinómico de cuarto grado: $y=ex^4+dx^3+cx^2+bx+a$.
- **configuration**: una cadena que especifica qué valores de regresión se incluyen, el orden en que aparecen y si se crean encabezados de fila o de columna. Introduzca los códigos separados por espacios y en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. Si no se especifica configuration, se utiliza una configuración predeterminada. H: crea encabezados de columna. h: crea encabezados de fila. sl: incluye la pendiente. Este código solo es válido para los modelos 1-6. int: incluye la ordenada en el origen. Este código solo es válido para los modelos 1-6. cor: incluye el coeficiente de correlación. Este código solo es válido para los modelos 1-6. cd: incluye el coeficiente de determinación. Este código solo es válido para los modelos 1-6 y 8-10. sCov: incluye la covarianza de la muestra. Este código solo es válido para los modelos 1-6. pCov: incluye la covarianza de la población. Este código solo es válido para los modelos 1-6. L: incluye el parámetro L. Este código solo es válido para el modelo 7. a: incluye el parámetro a. Este código solo es válido para los modelos 7-11. b: incluye el parámetro b. Este código solo es válido para los modelos 7-11. c: incluye el parámetro c. Este código solo es válido para los modelos 8-11. d: incluye el parámetro d. Este código solo es válido para los modelos 8 y 10-11. e: incluye el parámetro e. Este código solo es válido para el modelo 11. py: crea dos celdas: una para introducir un valor x y otra para mostrar el valor y pronosticado. px: crea dos celdas: una para introducir un valor y y otra para mostrar el valor x pronosticado.

0250 Función PredY

ID 250 · título recuperado de la página española

Sintaxis

PredY(mode, x, parameters)

Devuelve el valor y pronosticado para un valor x determinado utilizando el modelo de regresión y los coeficientes especificados.

Parámetros

- mode: especifica el modelo de regresión utilizado para calcular el valor y pronosticado. 1: modelo lineal: $y=sl*x+int$. 2: modelo logarítmico: $y=sl*\ln(x)+int$. 3: modelo exponencial: $y=int*\exp(sl*x)$. 4: modelo potencial: $y=int*x^{sl}$. 5: modelo exponencial de base variable: $y=int*sl^x$. 6: modelo recíproco: $y=sl/x+int$. 7: modelo logístico: $y=L/(1+a*\exp(b*x))$. 8: modelo sinusoidal: $y=a*\sin(b*x+c)+d$. 9: modelo cuadrático: $y=cx^2+bx+a$. 10: modelo cúbico: $y=dx^3+cx^2+bx+a$. 11: modelo polinómico de cuarto grado: $y=ex^4+dx^3+cx^2+bx+a$.
- x: el valor x para el que se calcula el valor y pronosticado.
- parameters: los coeficientes necesarios para el modelo de regresión seleccionado. Especifíquelos como una única lista o como argumentos separados, en el orden utilizado por el modelo.

Ejemplos

[Ejemplo gráfico E1] devuelve 57 para el modelo cuadrático $y=3x^2+2x+1$.
[Ejemplo gráfico E2] devuelve 11 para el modelo lineal $y=2x+3$.

0251 Función PredX

ID 251 · título recuperado de la página española

Sintaxis

PredX(mode, y, parameters)

Devuelve el valor x pronosticado para un valor y determinado utilizando el modelo de regresión y los coeficientes especificados.

Parámetros

- mode: especifica el modelo de regresión utilizado para calcular el valor x pronosticado. 1: modelo lineal: $y=sl*x+int$. 2: modelo logarítmico: $y=sl*\ln(x)+int$. 3: modelo exponencial: $y=int*\exp(sl*x)$. 4: modelo potencial: $y=int*x^{sl}$. 5: modelo exponencial de base variable: $y=int*sl^x$. 6: modelo recíproco: $y=sl/x+int$. 7: modelo logístico: $y=L/(1+a*\exp(b*x))$. 8: modelo sinusoidal: $y=a*\sin(b*x+c)+d$. 9: modelo cuadrático: $y=cx^2+bx+a$. 10: modelo cúbico: $y=dx^3+cx^2+bx+a$. 11: modelo polinómico de cuarto grado: $y=ex^4+dx^3+cx^2+bx+a$.
- y: el valor y para el que se calcula el valor x pronosticado.
- parameters: los coeficientes necesarios para el modelo de regresión seleccionado. Especifíquelos como una única lista o como argumentos separados, en el orden utilizado por el modelo.

Ejemplos

[Ejemplo gráfico E1] devuelve 4 para el modelo recíproco $y=8/x+1$.
[Ejemplo gráfico E2] devuelve 3 para el modelo exponencial de base variable $y=2*3^x$.

0252 Función HypZ1mean

ID 252 · título recuperado de la página española

Sintaxis

HypZ1mean(input_list [, configuration])
HypZ1mean(samp_mean, samp_size, null_pop_mean, pop_std_dev, sig_level, mode [, configuration])

Realiza una prueba Z de una muestra para la media de una población.

Parámetros

- input_list: los valores de entrada utilizados para realizar la prueba. La entrada puede ser un rango de celdas, una lista de referencias de celdas o una lista de valores. Los valores deben aparecer en el siguiente orden: samp_mean, samp_size, null_pop_mean, pop_std_dev, sig_level y mode.
- samp_mean: la media muestral.
- samp_size: el tamaño de la muestra. Debe ser un número entero positivo.
- null_pop_mean: la media poblacional especificada por la hipótesis nula.

- `pop_std_dev`: la desviación estándar de la población. Debe ser mayor que 0.
- `sig_level`: el nivel de significación utilizado en la prueba. Debe ser mayor que 0 y menor que 1.
- `mode`: especifica la hipótesis alternativa utilizada en la prueba. 1: menor que. 2: mayor que. 3: distinto de.
- `configuration`: una cadena que especifica qué resultados de la prueba se incluyen, el orden en que aparecen y si se crean encabezados. Introduzca los códigos separados por espacios en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. Una cadena vacía "" utiliza la configuración predeterminada, que incluye todos los resultados y encabezados. `h`: crea celdas de encabezado. `acc`: incluye 0 si se rechaza la hipótesis nula y 1 si no se rechaza. `tZ`: incluye el valor Z de la prueba. `tM`: incluye la media de la prueba. `prob`: incluye el valor p: de cola inferior para el modo 1, de cola superior para el modo 2 y bilateral para el modo 3. `cZ`: incluye el valor Z crítico. Para el modo 3, se devuelve el valor Z crítico negativo inferior. `cx1`: incluye el primer valor crítico ($\square$ 1). `cx2`: incluye el segundo valor crítico ($\square$ 2).

0253 Función HypZ2mean

ID 253 · título recuperado de la página española

Sintaxis

```
HypZ2mean(input_list [, configuration])
HypZ2mean(samp_mean, samp_mean2, samp_size, samp_size2, pop_std_dev, pop_std_dev2, sig_level, mode [, configuration])
```

Realiza una prueba Z de dos muestras para la diferencia entre dos medias poblacionales.

Parámetros

- `input_list`: los valores de entrada utilizados para realizar la prueba. La entrada puede ser un rango de celdas, una lista de referencias de celdas o una lista de valores. Los valores deben aparecer en el siguiente orden: `samp_mean`, `samp_mean2`, `samp_size`, `samp_size2`, `pop_std_dev`, `pop_std_dev2`, `sig_level` y `mode`.
- `samp_mean`: la media de la primera muestra.
- `samp_mean2`: la media de la segunda muestra.
- `samp_size`: el tamaño de la primera muestra.
- `samp_size2`: el tamaño de la segunda muestra.
- `pop_std_dev`: la desviación estándar poblacional de la primera muestra.
- `pop_std_dev2`: la desviación estándar poblacional de la segunda muestra.
- `sig_level`: el nivel de significación utilizado en la prueba.
- `mode`: especifica la hipótesis alternativa utilizada en la prueba. 1: menor que. 2: mayor que. 3: distinto de.
- `configuration`: una cadena que especifica qué resultados de la prueba se incluyen, el orden en que aparecen y si se crean encabezados. Introduzca los códigos separados por espacios en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. Una cadena vacía "" utiliza la configuración predeterminada, que incluye todos los resultados y encabezados. `h`: crea celdas de encabezado. `acc`: incluye 0 si se rechaza la hipótesis nula y 1 si no se rechaza. `tZ`: incluye el valor Z de la prueba. `tM`: incluye la diferencia entre las medias muestrales ($\Delta \square$), calculada como `samp_mean - samp_mean2`. `prob`: incluye el valor p: de cola inferior para el modo 1, de cola superior para el modo 2 y bilateral para el modo 3. `cZ`: incluye el valor Z crítico. `cx1`: incluye el primer valor límite para la diferencia entre las medias muestrales ($\Delta \square$). Para el modo 3, este es el valor inferior. `cx2`: incluye el segundo valor límite para la diferencia entre las medias muestrales ($\Delta \square$). Para el modo 3, este es el valor superior. Para los modos 1 y 2, es igual a `cx1`.

0254 Función HypZ1prop

ID 254 · título recuperado de la página española

Sintaxis

```
HypZ1prop(input_list [, configuration])
HypZ1prop(succ_count, samp_size, null_pop_prop, sig_level, mode [, configuration])
```

Realiza una prueba Z de una muestra para una proporción poblacional.

Parámetros

- `input_list`: los valores de entrada utilizados para realizar la prueba. La entrada puede ser un rango de celdas, una lista de referencias de celdas o una lista de valores. Los valores deben aparecer en el siguiente orden: `succ_count`, `samp_size`, `null_pop_prop`, `sig_level` y `mode`.
- `succ_count`: el número de éxitos en la muestra.
- `samp_size`: el tamaño de la muestra.
- `null_pop_prop`: la proporción poblacional especificada por la hipótesis nula.

- **sig_level**: el nivel de significación utilizado en la prueba.
- **mode**: especifica la hipótesis alternativa utilizada en la prueba. 1: menor que. 2: mayor que. 3: distinto de.
- **configuration**: una cadena que especifica qué resultados de la prueba se incluyen, el orden en que aparecen y si se crean encabezados. Introduzca los códigos separados por espacios en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. Una cadena vacía "" utiliza la configuración predeterminada, que incluye todos los resultados y encabezados. **h**: crea celdas de encabezado. **acc**: incluye 0 si se rechaza la hipótesis nula y 1 si no se rechaza. **tZ**: incluye el valor Z de la prueba. **tP**: incluye la proporción muestral de éxitos ($\square$). **prob**: incluye el valor p: de cola inferior para el modo 1, de cola superior para el modo 2 y bilateral para el modo 3. **cZ**: incluye el valor Z crítico. Para el modo 3, se devuelve el valor Z crítico negativo inferior. **cp1**: incluye el primer valor límite para la proporción muestral de éxitos ($\square$). Para el modo 3, este es el valor inferior. **cp2**: incluye el segundo valor límite para la proporción muestral de éxitos ($\square$). Para el modo 3, este es el valor superior. Para los modos 1 y 2, es igual a cp1.

0255 Función HypZ2prop

ID 255 · título recuperado de la página española

Sintaxis

```
HypZ2prop(input_list [, configuration])
HypZ2prop(succ_count1, succ_count2, samp_size1, samp_size2, sig_level, mode [, configuration])
```

Realiza una prueba Z de dos muestras para la diferencia entre dos proporciones poblacionales.

Parámetros

- **input_list**: los valores de entrada utilizados para realizar la prueba. La entrada puede ser un rango de celdas, una lista de referencias de celdas o una lista de valores. Los valores deben aparecer en el siguiente orden: succ_count1, succ_count2, samp_size1, samp_size2, sig_level y mode.
- **succ_count1**: el número de éxitos en la primera muestra.
- **succ_count2**: el número de éxitos en la segunda muestra.
- **samp_size1**: el tamaño de la primera muestra.
- **samp_size2**: el tamaño de la segunda muestra.
- **sig_level**: el nivel de significación utilizado en la prueba.
- **mode**: especifica la hipótesis alternativa utilizada en la prueba. 1: menor que. 2: mayor que. 3: distinto de.
- **configuration**: una cadena que especifica qué resultados de la prueba se incluyen, el orden en que aparecen y si se crean encabezados. Introduzca los códigos separados por espacios en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. Una cadena vacía "" utiliza la configuración predeterminada, que incluye todos los resultados y encabezados. **h**: crea celdas de encabezado. **acc**: incluye 0 si se rechaza la hipótesis nula y 1 si no se rechaza. **tZ**: incluye el valor Z de la prueba. **tP**: incluye la diferencia entre las proporciones muestrales de éxitos ($\Delta\square$). **p1**: incluye la proporción de éxitos en la primera muestra ($\square1$). **p2**: incluye la proporción de éxitos en la segunda muestra ($\square2$). **prob**: incluye el valor p: de cola inferior para el modo 1, de cola superior para el modo 2 y bilateral para el modo 3. **cZ**: incluye el valor Z crítico. Para el modo 3, se devuelve el valor Z crítico negativo inferior. **cp1**: incluye el primer valor límite para la diferencia entre las proporciones muestrales de éxitos ($\Delta\square$). Para el modo 3, este es el valor inferior. **cp2**: incluye el segundo valor límite para la diferencia entre las proporciones muestrales de éxitos ($\Delta\square$). Para el modo 3, este es el valor superior. Para los modos 1 y 2, es igual a cp1.

0256 Función HypT1mean

ID 256 · título recuperado de la página española

Sintaxis

```
HypT1mean(input_list [, configuration])
HypT1mean(samp_mean, samp_std_dev, samp_size, null_pop_mean, sig_level, mode [, configuration])
```

Realiza una prueba t de una muestra para una media poblacional.

Parámetros

- **input_list**: los valores de entrada utilizados para realizar la prueba. La entrada puede ser un rango de celdas, una lista de referencias de celdas o una lista de valores. Los valores deben aparecer en el siguiente orden: samp_mean, samp_std_dev, samp_size, null_pop_mean, sig_level y mode.
- **samp_mean**: la media muestral.
- **samp_std_dev**: la desviación estándar muestral.
- **samp_size**: el tamaño de la muestra.

- **null_pop_mean**: la media poblacional especificada por la hipótesis nula.
- **sig_level**: el nivel de significación utilizado en la prueba.
- **mode**: especifica la hipótesis alternativa utilizada en la prueba. 1: menor que. 2: mayor que. 3: distinto de.
- **configuration**: una cadena que especifica qué resultados de la prueba se incluyen, el orden en que aparecen y si se crean encabezados. Introduzca los códigos separados por espacios en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. Una cadena vacía "" utiliza la configuración predeterminada, que incluye todos los resultados y encabezados. **h**: crea celdas de encabezado. **acc**: incluye 0 si se rechaza la hipótesis nula y 1 si no se rechaza. **tT**: incluye el valor t de la prueba. **tM**: incluye la media de la prueba. **prob**: incluye el valor p: de cola inferior para el modo 1, de cola superior para el modo 2 y bilateral para el modo 3. **df**: incluye los grados de libertad. **cT**: incluye el valor t crítico. Para el modo 3, se devuelve el valor t crítico negativo inferior. **cX**: incluye el valor crítico de la media asociado con el valor t crítico. Para el modo 3, se devuelve el valor crítico inferior de la media.

0257 Función HypT2mean

ID 257 · título recuperado de la página española

Sintaxis

```
HypT2mean(input_list [, configuration])
HypT2mean(samp_mean1, samp_mean2, samp_std_dev1, samp_std_dev2, samp_size1, samp_size2, pooled, sig_level, mode [, configuration])
```

Realiza una prueba t de dos muestras para la diferencia entre dos medias poblacionales.

Parámetros

- **input_list**: los valores de entrada utilizados para realizar la prueba. La entrada puede ser un rango de celdas, una lista de referencias de celdas o una lista de valores. Los valores deben aparecer en el siguiente orden: **samp_mean1**, **samp_mean2**, **samp_std_dev1**, **samp_std_dev2**, **samp_size1**, **samp_size2**, **pooled**, **sig_level** y **mode**.
- **samp_mean1**: la media de la primera muestra.
- **samp_mean2**: la media de la segunda muestra.
- **samp_std_dev1**: la desviación estándar de la primera muestra.
- **samp_std_dev2**: la desviación estándar de la segunda muestra.
- **samp_size1**: el tamaño de la primera muestra.
- **samp_size2**: el tamaño de la segunda muestra.
- **pooled**: especifica si se combinan las varianzas muestrales. 0: no combina las varianzas muestrales. 1: combina las varianzas muestrales.
- **sig_level**: el nivel de significación utilizado en la prueba.
- **mode**: especifica la hipótesis alternativa utilizada en la prueba. 1: menor que. 2: mayor que. 3: distinto de.
- **configuration**: una cadena que especifica qué resultados de la prueba se incluyen, el orden en que aparecen y si se crean encabezados. Introduzca los códigos separados por espacios en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. Una cadena vacía "" utiliza la configuración predeterminada, que incluye todos los resultados y encabezados. **h**: crea celdas de encabezado. **acc**: incluye 0 si se rechaza la hipótesis nula y 1 si no se rechaza. **tT**: incluye el valor t de la prueba. **tM**: incluye la diferencia entre las medias muestrales ($\Delta\bar{x}$), calculada como **samp_mean1** – **samp_mean2**. **prob**: incluye el valor p: de cola inferior para el modo 1, de cola superior para el modo 2 y bilateral para el modo 3. **df**: incluye los grados de libertad. **cT**: incluye el valor t crítico. Para el modo 3, se devuelve el valor t crítico negativo inferior. **cdx**: incluye la diferencia entre las medias muestrales ($\Delta\bar{x}$) correspondiente al valor t crítico. Para el modo 3, se devuelve el valor inferior.

0258 Función ConfZ1mean

ID 258 · título recuperado de la página española

Sintaxis

```
ConfZ1mean(input_list [, configuration])
ConfZ1mean(samp_mean, samp_size, pop_std_dev, conf_level [, configuration])
```

Calcula un intervalo de confianza normal de una muestra para una media poblacional.

Parámetros

- **input_list**: los valores de entrada utilizados para calcular el intervalo de confianza. La entrada puede ser un rango de celdas, una lista de referencias de celdas o una lista de valores. Los valores deben aparecer en el siguiente orden: **samp_mean**, **samp_size**, **pop_std_dev** y **conf_level**.

- `samp_mean`: la media muestral.
- `samp_size`: el tamaño de la muestra.
- `pop_std_dev`: la desviación estándar poblacional.
- `conf_level`: el nivel de confianza utilizado para calcular el intervalo.
- `configuration`: una cadena que especifica qué resultados se incluyen, el orden en que aparecen y si se crean encabezados. Introduzca los códigos separados por espacios en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. Una cadena vacía "" utiliza la configuración predeterminada, que incluye todos los resultados y encabezados. `h`: crea celdas de encabezado. `Z`: incluye el valor Z crítico negativo inferior. `low`: incluye el límite inferior del intervalo de confianza. `up`: incluye el límite superior del intervalo de confianza.

0259 Función ConfZ2mean

ID 259 · título recuperado de la página española

Sintaxis

```
ConfZ2mean(input_list [, configuration])
ConfZ2mean(samp_mean1, samp_mean2, samp_size1, samp_size2, pop_std_dev1, pop_std_dev2, conf_level [, configuration])
```

Calcula un intervalo de confianza normal de dos muestras para la diferencia entre dos medias poblacionales.

Parámetros

- `input_list`: los valores de entrada utilizados para calcular el intervalo de confianza. La entrada puede ser un rango de celdas, una lista de referencias de celdas o una lista de valores. Los valores deben aparecer en el siguiente orden: `samp_mean1`, `samp_mean2`, `samp_size1`, `samp_size2`, `pop_std_dev1`, `pop_std_dev2` y `conf_level`.
- `samp_mean1`: la media de la primera muestra.
- `samp_mean2`: la media de la segunda muestra.
- `samp_size1`: el tamaño de la primera muestra.
- `samp_size2`: el tamaño de la segunda muestra.
- `pop_std_dev1`: la desviación estándar poblacional correspondiente a la primera muestra.
- `pop_std_dev2`: la desviación estándar poblacional correspondiente a la segunda muestra.
- `conf_level`: el nivel de confianza utilizado para calcular el intervalo.
- `configuration`: una cadena que especifica qué resultados se incluyen, el orden en que aparecen y si se crean encabezados. Introduzca los códigos separados por espacios en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. Una cadena vacía "" utiliza la configuración predeterminada, que incluye todos los resultados y encabezados. `h`: crea celdas de encabezado. `Z`: incluye el valor Z crítico negativo inferior. `low`: incluye el límite inferior del intervalo de confianza. `up`: incluye el límite superior del intervalo de confianza.

0260 Función ConfZ1prop

ID 260 · título recuperado de la página española

Sintaxis

```
ConfZ1prop(input_list [, configuration])
ConfZ1prop(succ_count, samp_size, conf_level [, configuration])
```

Calcula un intervalo de confianza normal de una muestra para una proporción poblacional.

Parámetros

- `input_list`: los valores de entrada utilizados para calcular el intervalo de confianza. La entrada puede ser un rango de celdas, una lista de referencias de celdas o una lista de valores. Los valores deben aparecer en el siguiente orden: `succ_count`, `samp_size` y `conf_level`.
- `succ_count`: el número de éxitos en la muestra.
- `samp_size`: el tamaño de la muestra.
- `conf_level`: el nivel de confianza utilizado para calcular el intervalo.
- `configuration`: una cadena que especifica qué resultados se incluyen, el orden en que aparecen y si se crean encabezados. Introduzca los códigos separados por espacios en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. Una cadena vacía "" utiliza la configuración predeterminada, que incluye todos los resultados y encabezados. `h`: crea celdas de encabezado. `Z`: incluye el valor Z crítico negativo inferior. `low`: incluye el límite inferior del intervalo de confianza. `up`: incluye el límite superior del intervalo de confianza.

0261 Función ConfZ2prop

ID 261 · título recuperado de la página española

Sintaxis

```
ConfZ2prop(input_list [, configuration])  
ConfZ2prop(succ_count1, succ_count2, samp_size1, samp_size2, conf_level [, configuration])
```

Calcula un intervalo de confianza normal de dos muestras para la diferencia entre dos proporciones poblacionales.

Parámetros

- **input_list**: los valores de entrada utilizados para calcular el intervalo de confianza. La entrada puede ser un rango de celdas, una lista de referencias de celdas o una lista de valores. Los valores deben aparecer en el siguiente orden: succ_count1, succ_count2, samp_size1, samp_size2 y conf_level.
- **succ_count1**: el número de éxitos en la primera muestra.
- **succ_count2**: el número de éxitos en la segunda muestra.
- **samp_size1**: el tamaño de la primera muestra.
- **samp_size2**: el tamaño de la segunda muestra.
- **conf_level**: el nivel de confianza utilizado para calcular el intervalo.
- **configuration**: una cadena que especifica qué resultados se incluyen, el orden en que aparecen y si se crean encabezados. Introduzca los códigos separados por espacios en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. Una cadena vacía "" utiliza la configuración predeterminada, que incluye todos los resultados y encabezados. h: crea celdas de encabezado. Z: incluye el valor Z crítico negativo inferior. low: incluye el límite inferior del intervalo de confianza. up: incluye el límite superior del intervalo de confianza.

0262 Función ConfT1mean

ID 262 · título recuperado de la página española

Sintaxis

```
ConfT1mean(input_list [, configuration])  
ConfT1mean(samp_mean, samp_std_dev, samp_size, conf_level [, configuration])
```

Calcula un intervalo de confianza t de Student de una muestra para una media poblacional.

Parámetros

- **input_list**: los valores de entrada utilizados para calcular el intervalo de confianza. La entrada puede ser un rango de celdas, una lista de referencias de celdas o una lista de valores. Los valores deben aparecer en el siguiente orden: samp_mean, samp_std_dev, samp_size y conf_level.
- **samp_mean**: la media muestral.
- **samp_std_dev**: la desviación estándar muestral.
- **samp_size**: el tamaño de la muestra.
- **conf_level**: el nivel de confianza utilizado para calcular el intervalo.
- **configuration**: una cadena que especifica qué resultados se incluyen, el orden en que aparecen y si se crean encabezados. Introduzca los códigos separados por espacios en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. Una cadena vacía "" utiliza la configuración predeterminada, que incluye todos los resultados y encabezados. h: crea celdas de encabezado. df: incluye los grados de libertad. T: incluye el valor t crítico negativo inferior. low: incluye el límite inferior del intervalo de confianza. up: incluye el límite superior del intervalo de confianza.

0263 Función ConfT2mean

ID 263 · título recuperado de la página española

Sintaxis

```
ConfT2mean(input_list [, configuration])  
ConfT2mean(samp_mean1, samp_mean2, samp_std_dev1, samp_std_dev2, samp_size1, samp_size2, pooled, conf_level [, configuration])
```

Calcula un intervalo de confianza t de Student de dos muestras para la diferencia entre dos medias poblacionales.

Parámetros

- **input_list**: los valores de entrada utilizados para calcular el intervalo de confianza. La entrada puede ser un rango de celdas, una lista de referencias de celdas o una lista de valores. Los valores deben aparecer en el siguiente orden: **samp_mean1**, **samp_mean2**, **samp_std_dev1**, **samp_std_dev2**, **samp_size1**, **samp_size2**, **pooled** y **conf_level**.
- **samp_mean1**: la media de la primera muestra.
- **samp_mean2**: la media de la segunda muestra.
- **samp_std_dev1**: la desviación estándar de la primera muestra.
- **samp_std_dev2**: la desviación estándar de la segunda muestra.
- **samp_size1**: el tamaño de la primera muestra.
- **samp_size2**: el tamaño de la segunda muestra.
- **pooled**: especifica si se combinan las varianzas muestrales. 0: no combina las varianzas muestrales. 1: combina las varianzas muestrales.
- **conf_level**: el nivel de confianza utilizado para calcular el intervalo.
- **configuration**: una cadena que especifica qué resultados se incluyen, el orden en que aparecen y si se crean encabezados. Introduzca los códigos separados por espacios en el orden en que deban aparecer los valores correspondientes en la hoja de cálculo. Una cadena vacía "" utiliza la configuración predeterminada, que incluye todos los resultados y encabezados. **h**: crea celdas de encabezado. **df**: incluye los grados de libertad. **T**: incluye el valor t crítico negativo inferior. **low**: incluye el límite inferior del intervalo de confianza. **up**: incluye el límite superior del intervalo de confianza.

1724 CellHasData — Devuelve el número de celdas que contienen datos en el conjunto de celdas designado.

ID 1724 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CellHasData

Explicación española: Devuelve el número de celdas que contienen datos en el conjunto de celdas designado.

Sintaxis

```
CellHasData()
CellHasData(Row, Column)
CellHasData(Row_1, Column_1, Row_2, Column_2)
```

CellHasData devuelve el número de celdas que contienen datos en el conjunto de celdas designado.

Si el comando no recibe parámetros, se contarán todas las celdas que contienen datos.

Si se especifica una sola fila **Row** y columna **Column**, solo se contará la celda en esa ubicación.

Si se especifica una segunda fila y columna, se contarán todas las celdas en un área rectangular donde la esquina superior izquierda es la celda en **Row_1**, **Column_1** y la esquina inferior derecha es la celda en **Row_2**, **Column_2**.

1722 ClearCell — () borra todas las celdas en la hoja de cálculo.

ID 1722 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ClearCell

Explicación española: () borra todas las celdas en la hoja de cálculo.

Sintaxis

```
ClearCell()
ClearCell(Row, Column)
ClearCell(Row_1, Column_1, Row_2, Column_2)
```

ClearCell() borra todas las celdas en la hoja de cálculo.

Si se especifica la fila **Row** y la columna **Column**, se borrará una sola celda.

Si se especifica una segunda fila y columna, se borrarán todas las celdas en un área rectangular donde la esquina superior izquierda es la celda en **Row_1**, **Column_1** y la esquina inferior derecha es la celda en **Row_2**, **Column_2**.

0264 Aplicación Var 1 estadística

ID 264 · título localizado en HelpEs

La aplicación Var 1 estadística puede almacenar hasta diez conjuntos de datos al mismo tiempo. Puede realizar análisis estadístico de una variable en uno o varios conjuntos de datos.

La aplicación Var 1 estadística se inicia con la vista numérica, que se usa para introducir datos. La vista simbólica se usa para definir un análisis especificando qué columna contiene los datos y qué columna, si la hay, contiene las frecuencias.

La vista de gráfico se usa para mostrar gráficos estadísticos de datos de 1 variable, incluidos histogramas, diagramas de caja y bigotes, gráficos de cuantiles normales y otros tipos de gráficos.

Para iniciar la aplicación Var 1 estadística, vaya a la biblioteca de aplicaciones y toque el icono de la aplicación Var 1 estadística. También puede utilizar la rueda de selección para seleccionar el icono de la aplicación Var 1 estadística y, a continuación, toque **[[Inicio]]** o presione Enter para iniciar la aplicación.

0265 Vista simbólica de Var 1 estadística

ID 265 · título localizado en HelpEs

Presione Symb para abrir la Vista simbólica. Puede definir hasta cinco análisis de 1 variable (H1-H5) mediante la selección de una columna de datos y una columna de frecuencia opcional para cada análisis de datos. Para la columna de datos, puede introducir el nombre de una columna (D0-D9) o una expresión matemática relacionada con el nombre de una columna (por ejemplo, D1-9.8). También existe un campo Gráfico para cada análisis en el que puede seleccionar la representación gráfica que mejor se adapte a sus objetivos. Las opciones gráficas incluyen:

- Histogramas
- Gráficos de diagramas de caja, con y sin valores atípicos
- Gráficos de probabilidad normal
- Gráficos de línea
- Gráficos de barras
- Diagramas de Pareto
- Diagramas de control
- Gráficos de puntos
- Gráficos de tallo y hojas, con tallos únicos (grupos de 10/100/1000) o divididos (grupos de 5/50/500 etc.)
- Diagrama de tortas

Los botones del menú son:

- **[[Editar]]** permite editar el valor seleccionado
- **[[Selec.]]** permite seleccionar el tipo de gráfico o el color del gráfico
- **[[✓]]** permite convertir un análisis en activo o inactivo
- **[[Columna]]** seleccione el nombre de una columna de la Vista numérica
- **[[Mostr.]]** muestra la ecuación de ajuste en modo de pantalla completa con el desplazamiento horizontal y vertical activado
- **[[Eval.]]** evalúa la expresión resaltada y resuelve todas las referencias a otras definiciones

Cada conjunto de datos activo (H1-H5) se utilizará con fines de creación de gráficos en la Vista de gráfico, así como para mostrar resúmenes de estadísticas en la Vista numérica al tocar **[[Estads.]]**.

0266 Vista de gráfico de Var 1 estadística

ID 266 · título localizado en HelpEs

Presione Plot para acceder a la Vista de gráfico. Esta vista muestra los gráficos estadísticos de 1 variable para los análisis activos (H1-H5). El menú es similar al de la Vista de gráfico de Función, con la posibilidad de acercar, alejar y realizar trazados. Toque **[[Menú]]** para activar y desactivar el menú.

Los botones del menú son:

- **[[Zoom]]** abre el menú Zoom, con opciones para acercar, alejar, etc.
- **[[Trazar]]** permite activar y desactivar el cursor de trazado
- **[[Defn]]** muestra la definición de la función en la que se realizará el trazado
- **[[Menú]]** activa o desactiva el menú

Utilice la rueda hacia la izquierda/derecha para realizar un trazado a lo largo de una gráfica estadística. Utilice la rueda hacia arriba/abajo para cambiar de un gráfico a otro. Presione + para acercar la posición actual del cursor y presione - para alejarla. Ajuste el factor de zoom en el menú **[[Zoom]]**.

Todos los gestos comunes a las Vistas de gráfico son compatibles aquí también. Consulte Vista de gráfico para obtener más detalles.

0267 Config. de gráfico de Var 1 estadística

ID 267 · título localizado en HelpEs

Presione Shift Plot para abrir Config. de gráfico de Var 1 estadística. La página 1 de Config. de gráfico contiene configuraciones que controlan la apariencia de los gráficos estadísticos de 1 variable.

Los campos son los siguientes:

- Anch H: ancho del intervalo para histogramas
- Rng H: rango de datos que desea trazar.
- Rng X: el rango horizontal de la ventana de gráfico
- Rng Y: el rango vertical de la ventana de gráfico
- Mrc X: espaciado entre marcas de graduación horizontal
- Mrc Y: espaciado entre marcas de graduación vertical

Los botones del menú en la primera página son:

- [[Editar]] abre un nuevo cuadro de edición para editar el valor que aparece en el campo seleccionado.
- [-Página 1/2 ▼-] muestra la segunda página de la configuración

Pulse [-Página 1/2 ▼-] para ver la segunda página de la configuración. Aquí, los campos son:

- Ejes: activa o desactiva los ejes
- Etiquetas: activa o desactiva las etiquetas del eje
- Puntos cuadr.: activa o desactiva los puntos de la cuadrícula
- Líneas cuadríc.: activa o desactiva las líneas de la cuadrícula
- Cursor: permite elegir entre cursores estándar, invertido y parpadeante

Los botones del menú en la segunda página son:

- [[✓]] activa o desactiva la configuración actual
- [[Selec.]] permite elegir de un cuadro de elección
- [-▲ Página 2/2-] retorna a la primera página de la configuración

0268 Vista Numérica de Var 1 estadística

ID 268 · título localizado en HelpEs

Presione Num para volver a esta vista en cualquier momento. Esta vista contiene una tabla con hasta diez columnas de datos, denominadas de D1 a D9 y D0.

Los botones del menú son:

- [[Editar]] abre una línea de entrada para editar el valor deseado
- [[Más]] abre un menú con opciones para editar la lista
- [[Ir a]] pasa a un elemento específico en la lista. Útil para listas muy grandes.
- [[Orden.]] organiza la columna actual en orden ascendente o descendente
- [[Crear]] genera una columna de datos basada en una fórmula algebraica
- [[Estado]] proporciona resúmenes de estadísticas de los análisis definidos actualmente (consulte Vista simbólica)

Introduzca sus datos de forma manual o almacene los datos de la lista en D1, D2, etc. Utilice la función [[Make]] para crear los datos basados en una fórmula algebraica. También puede pegar los datos copiados de otra aplicación o de los Editores de lista y matriz.

También puede poner un nombre a cada columna de datos. Toque en el encabezado de la columna y luego edite el nombre o comience a escribir para introducir un nuevo nombre.

El menú [[Más]] contiene las siguientes opciones para editar una lista:

- Insertar Fila: inserta una nueva fila en la lista actual. La nueva fila contiene 0 como su elemento.
- Eliminar Columna: borra el contenido de la lista actual. Para eliminar un solo elemento, selecciónelo y pulse la tecla Retroceso.

- **Seleccionar Fila:** selecciona la fila actual. Una vez seleccionada, la fila se puede copiar. **Columna:** selecciona la lista actual. Una vez seleccionada, la lista se puede copiar. **Cuadro:** abre un cuadro de diálogo para seleccionar una matriz rectangular definida por un punto de partida y una ubicación final. También puede mantener pulsada una celda para iniciar la selección, y a continuación, arrastrar para seleccionar una matriz rectangular de elementos. Una vez seleccionada, la matriz se puede copiar. **Intercambiar final:** intercambia las celdas de inicio y final de la matriz de celdas rectangular seleccionada.
- **Selección:** activa y desactiva el modo de selección. También puede mantener pulsada una celda, y a continuación, arrastrar para seleccionar.
- **Interc Columna:** intercambia el contenido de dos columnas (listas).

1637 Crear columna

ID 1637 · título localizado en HelpEs

El asistente de datos de Crear columna es básicamente un método abreviado para usar el comando MAKELIST y almacenar los resultados en la Vista numérica. Los campos de este asistente son:

- **Expresión:** introduzca la expresión de generación para su lista de números reales
- **Var:** declare la variable independiente de su expresión. Todas las demás variables de su expresión se tomarán como constantes. Tenga en cuenta que su variable puede ser simulada; es decir, es posible que no aparezca en su expresión en absoluto.
- **Inicio:** ingrese el valor inicial para su variable
- **Final:** ingrese el valor final de su variable
- **Paso:** ingrese el valor de incremento para su variable
- **Col:** una lista desplegable para elegir el destino de su lista de números reales

Complete los campos y toque **[[OK]]** para generar la columna de números y guárdelos en la lista que especificó, o toque **[[Cancelar]]** para volver a la Vista numérica sin crear una lista. Use este asistente para crear con facilidad una lista de enteros aleatorios, una distribución de muestra, etc.

0269 Variables de Var 1 estadística

ID 269 · título localizado en HelpEs

Para mostrar las variables relacionadas con la aplicación Var 1 estadística, presione Vars, toque **[[Apl.]]** y seleccione Var 1 estadística.

La aplicación Var 1 estadística tiene las siguientes variables:

- **Resultados** (consulte a continuación)
- **Simbólica** (consulte a continuación)
- **Gráfico** (consulte a continuación)
- **Númerica** (consulte a continuación)
- **Modos** (consulte Variables de aplicaciones comunes más arriba)

0270 Variables de resultados

ID 270 · título localizado en HelpEs

Las variables de la aplicación Var 1 estadística guardan los resultados de los cálculos efectuados al pulsar el botón **[[Estado]]** en la Vista numérica de la aplicación o al ejecutar el comando Do1VStats.

0271 Variable de aplicación Nbltem

ID 271 · título recuperado de la página española

Nbltem : contiene el número de puntos de datos en el análisis de 1 variable actual (H1-H5).

0272 Variable de aplicación MinVal

ID 272 · título recuperado de la página española

MinVal : contiene el valor mínimo de los datos definidos en el análisis de 1 variable actual (H1-H5).

0273 Variable de aplicación Q₁

ID 273 · título recuperado de la página española

Q_1 : el valor del primer cuartil en el análisis actual de 1 variable (H1-H5).

0274 Variable de aplicación MedVal

ID 274 · título recuperado de la página española

MedVal : contiene la mediana del análisis de 1 variable actual (H1-H5).

0275 Variable de la aplicación Q_3

ID 275 · título recuperado de la página española

Q_3 : el valor del tercer cuartil en el análisis actual de 1 variable (H1-H5).

0276 Variable de aplicación MaxVal

ID 276 · título recuperado de la página española

MaxVal : el valor máximo en el análisis actual de 1 variable (H1-H5).

0277 Variable de aplicación ΣX

ID 277 · título recuperado de la página española

ΣX : La suma del conjunto de datos en el análisis de 1 variable actual (H1-H5).

0278 Variable de aplicación ΣX^2

ID 278 · título recuperado de la página española

ΣX^2 : La suma de los cuadrados del conjunto de datos en el análisis de 1 variable actual (H1-H5).

0279 Variable de aplicación MeanX

ID 279 · título recuperado de la página española

MeanX : la media de los datos definidos en el análisis de 1 variable actual (H1-H5).

0280 Variable de aplicación sX

ID 280 · título recuperado de la página española

sX : contiene la desviación estándar de muestra del conjunto de datos del análisis de 1 variable actual (H1-H5).

0281 σX — La desviación estándar de la población del conjunto de datos en el análisis de 1 variable actual (H1-H5).

ID 281 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: σX

Explicación española: La desviación estándar de la población del conjunto de datos en el análisis de 1 variable actual (H1-H5).

σX : La desviación estándar de la población del conjunto de datos en el análisis de 1 variable actual (H1-H5).

0282 Variable de aplicación serrX

ID 282 · título recuperado de la página española

serrX : El error estándar del conjunto de datos en el análisis actual de 1 variable (H1-H5).

0283 Variable de aplicación ssX

ID 283 · título recuperado de la página española

ssX : la suma de las desviaciones al cuadrado de x a partir de la media de x del conjunto de datos en el análisis de 1 variable (H1-H5) actual.

0284 Variables simbólicas

ID 284 · título localizado en HelpEs

Las variables simbólicas de Var 1 estadística son H1-H5. Estas variables contienen los valores de los datos de un análisis estadístico de 1 variable. Por ejemplo, H1(n) devuelve el valor n-ésimo del conjunto de datos para el análisis H1. Sin un argumento, H1 devuelve una lista de los objetos que definen H1. Estos objetos son los siguientes, en el orden indicado:

- Una cadena o expresión (entre comillas simples) que define la lista de datos
- Una cadena o expresión (entre comillas simples) que define opcionalmente las frecuencias para cada uno de los valores en la lista de datos
- El número de tipo de gráfico
- El número de opción
- El color del gráfico

El número de tipo de gráfico es un número entero del 1-9 que controla qué tipo de gráfico estadístico se utiliza con cada una de las variables H1- H5. La correspondencia se muestra a continuación.

1 Histogram (Histograma) (predeterminado)

2 Box and Whisker (Diagrama de caja)

3 Normal Probability (Probabilidad normal)

4 Line (Líneas)

5 Bar (Barras)

6 Pareto (Pareto)

7 Control

8 Punto

9 Tallo y hojas

El número de opción es un número entero de 0-2 que controla cualquier opción disponible para el tipo de gráfico. La correspondencia se muestra a continuación.

0 No hay opción

1 No mostrar valores atípicos para el diagrama de caja

2 Mostrar valores atípicos para el diagrama de caja

Ejemplo

H3:={"D1", "", 2, 1, #FF:24h} define H3 para utilizar D1 para su lista de datos, sin frecuencias, y dibujar un gráfico de diagrama de caja sin valores atípicos utilizando un color azul.

0285 Variables numéricas

ID 285 · título localizado en HelpEs

Las variables numéricas de Var 1 estadística son de D1 a D9 y D0. Cada una representa un único conjunto de datos y contiene los valores de ese conjunto de datos. Todas son variables de lista y son compatibles con las variables de lista de Var 2 estadística C0-C9, así como las variables de Inicio L0-L9.

0286 Variables de gráfico

ID 286 · título localizado en HelpEs

Además de las variables de la Vista de gráfico comunes (consulte Variables de aplicaciones comunes más arriba), la aplicación Var 1 estadística incluye tres variables específicas de la aplicación:

- Hmin: valor mínimo que se va a incluir en el histograma
- Hmax: valor máximo que se va a incluir en el histograma
- Hwidth: ancho de la barra del histograma (ancho del intervalo)

0287 Variable Hwidth

ID 287 · título recuperado de la página española

Hwidth : el ancho de una barra para gráficos del histograma (ancho del contenedor).

0288 Variable Hmin

ID 288 · título recuperado de la página española

Hmin : el valor mínimo que se incluirá en el histograma.

0289 Variable Hmax

ID 289 · título recuperado de la página española

Hmax : el valor máximo que se incluirán en el histograma.

0290 Funciones de aplicación Var 1 estadística

ID 290 · título localizado en HelpEs

La aplicación Var 1 estadística tiene 3 funciones diseñadas de manera que funcionen conjuntamente para calcular resúmenes de estadísticas basados en uno de los análisis estadísticos (H1-H5) definidos en la Vista simbólica de la aplicación Var 1 estadística.

0291 Función de aplicación Do1VStats

ID 291 · título recuperado de la página española

Sintaxis

Do1VStats(Hn)

Efectúa los mismos cálculos que cuando se toca [[Estado]] en la Vista numérica de la aplicación Var 1 estadística, y guarda los resultados en las variables de resultados de Var 1 estadística correspondientes. Hn debe corresponder a una de las variables H1-H5 de la Vista simbólica de la aplicación Var 1 estadística.

0292 Función de aplicación SetFreq

ID 292 · título recuperado de la página española

Sintaxis

SetFreq(Hn, Dn)
SetFreq(Hn, Num)

Establece la frecuencia. La sintaxis puede ser SetFreq(Hn, Dn) o SetFreq(Hn, Num). Configura la frecuencia de uno de los análisis estadísticos (H1-H5) definidos en la Vista simbólica de la aplicación Var 1 estadística. La frecuencia puede ser una de las variables de columna D0-D9 o cualquier número entero positivo. Hn debe corresponder a una de las variables H1-H5 de la Vista simbólica de la aplicación Var 1 estadística. Si se utiliza, Dn debe ser una de las variables de columna D0-D9; de lo contrario, el valor debe ser un entero positivo.

Ejemplos

[Ejemplo gráfico E1] configura la frecuencia para cada valor en el conjunto de datos para análisis H1 en 7.
[Ejemplo gráfico E2] configura las frecuencias del conjunto de datos en H3 en la columna D3.

0293 Función de aplicación SetSample

ID 293 · título recuperado de la página española

Sintaxis

SetSample(Hn, Dn)

Configura los datos de muestra. Configura los datos de muestra para uno de los análisis estadísticos (H1-H5) definidos en la Vista simbólica de la aplicación Var 1 estadística en una de las variables de columna D0-D9.

Ejemplos

[Ejemplo gráfico E1] configura el análisis H2 para que utilice la columna D4 para sus datos.

0294 Función de aplicación CHECK

ID 294 · título recuperado de la página española

Sintaxis

CHECK(n)

Marca (selecciona) la definición correspondiente en la Vista simbólica. El entero n debe estar entre 0 y 5.

0295 Función de aplicación UNCHECK

ID 295 · título recuperado de la página española

Sintaxis

UNCHECK(n)

Desactiva (anula la selección de) la definición correspondiente en la Vista simbólica. El entero n debe ser un valor entre 0 y 5.

0296 Función de aplicación ISCHECK

ID 296 · título recuperado de la página española

Sintaxis

ISCHECK(n)

Devuelve 1 o 0 dependiendo de si la definición correspondiente en la Vista simbólica está seleccionada o no. El entero n debe estar entre 0 y 5.

0297 Aplicación Var 2 estadística

ID 297 · título localizado en HelpEs

La aplicación 2-Var estadística puede almacenar hasta diez conjuntos de datos simultáneamente. Puede realizar análisis estadísticos de dos variables de uno o varios conjuntos de datos.

La aplicación 2-Var estadística se inicia con la Vista numérica, que se utiliza para introducir datos. La Vista simbólica se utiliza para especificar las columnas que contienen datos.

Para iniciar la aplicación Var 2 estadística, vaya a la biblioteca de aplicaciones y pulse el icono de la aplicación Var 2 estadística. También puede utilizar la rueda para seleccionar el icono de la aplicación Var 2 estadística, luego pulse [[Inicio]] o presione Enter para iniciar la aplicación.

0298 Vista simbólica de Var 2 estadística

ID 298 · título localizado en HelpEs

Presione Symb para abrir la Vista simbólica de 2-Var estadística. Se pueden definir hasta cinco análisis de 2 variables, denominadas de S1 a S5, con diferentes marcadores para cada gráfico de dispersión.

Cada definición de conjunto de datos incluye los siguientes campos:

- Sn: define las columnas independientes y dependientes, así como una columna de frecuencia opcional para los datos de la columna dependiente. También puede seleccionar un tipo de punto y un color para el gráfico de dispersión.
- Tipo: selecciona un tipo de función para ajustarse a los datos
- Marca: permite seleccionar un estilo para las marcas de punto de datos de cada gráfico de dispersión
- Fit: contiene la ecuación de su ajuste, así como un selector de color a la izquierda del ajuste para que pueda elegir un color para el gráfico de ajuste.

Para las columnas independientes y dependientes, puede introducir expresiones matemáticas en términos de un nombre de columna (por ejemplo, 2-C1).

Cada conjunto de datos activo (S1-S5) se utilizará con fines de creación de gráficos en la Vista de gráfico, así como para mostrar resúmenes de estadísticas en la Vista numérica al tocar [[Estads]].

Los botones del menú son:

- [[Editar]] permite editar el valor seleccionado
- [[Selec.]] permite seleccionar el tipo de gráfico o el color del gráfico
- [[✓]] permite convertir un análisis en activo o inactivo
- [[Columna]] un cuadro de elección para seleccionar el nombre de una columna
- [[Ajuste]] activa o desactiva el ajuste en la Vista de gráfico
- [[Mostr.]] muestra la ecuación de ajuste en modo de pantalla completa con el desplazamiento horizontal y vertical activado
- [[Eval.]] evalúa la expresión resaltada y resuelve todas las referencias a otras definiciones

0299 Vista de gráfico de Var 2 estadística

Presione Plot para acceder a la Vista de gráfico de 2-Var estadística. En esta vista se muestran los gráficos de dispersión de los análisis activos. Toque [[Menú]] para activar y desactivar el menú.

Los botones del menú son:

- [[Zoom]] permite acercar o alejar los gráficos
- [[Trazar]] permite activar y desactivar el trazado
- [[Ir a]] abre un formulario de entrada para colocar el marcador en un valor x específico
- [[Ajst]] muestra ajustes de cada gráfico de dispersión
- [[Func.]] abre el menú de Funciones Ajuste: es un duplicado de la tecla del menú Ajuste descrito antes Esbozo: haga un boceto de su propio ajuste con su dedo (vea a continuación) Defn: muestra la definición del gráfico actual que se está trazando
- [[Menú]] activa o desactiva el menú

Al seleccionar la opción Boceto usted vuelve a la Vista de gráfico, con un mensaje en la parte inferior de la pantalla para que haga un boceto del ajuste de una función con su dedo. Haga un boceto de su ajuste con su dedo; toque [[OK]] si le gusta su boceto, o basta hacer un boceto de un nuevo ajuste si no le gusta el original. Cuando tenga el ajuste que quiere, toque [[OK]]. El tipo de ajuste del conjunto de datos actual en la Vista simbólica (S1-S5) cambiará a Definido por el usuario y la expresión (en X) de su ajuste se guardará como el ajuste definido por el usuario.

Seleccione Definición para ver la expresión del gráfico actual. En el cuadro de definición, puede pulsar [[Editar]] para editar la expresión; cuando haya terminado de editar, pulse [[OK]] para ver el cambio en el gráfico. También puede pulsar Transformar para manipular el gráfico actual. Pulse [[OK]] cuando haya terminado para regresar al cuadro de definición. Pulse de nuevo el botón de menú [[↵]] para cerrar el cuadro.

Utilice la rueda hacia la izquierda/derecha para realizar un trazado a lo largo de un diagrama de dispersión o un ajuste. Utilice la rueda hacia arriba/abajo para moverse de un diagrama de dispersión al siguiente, o del diagrama de dispersión hasta el ajuste.

Presione + para acercar la posición actual del cursor y pulse - para alejarla. Ajuste el factor de zoom en el menú [[Zoom]].

Todos los gestos comunes a las Vistas de gráfico son compatibles aquí también. Consulte Vista de gráfico para obtener más detalles.

0300 Config. de gráfico de Var 2 estadística

Presione Shift Plot para abrir la página 1 de Config. de gráfico de Var 2 estadística. Esta vista es similar a la página 1 de Config. de gráfico de Var 1 estadística.

En la primera página, los campos son:

- Rng X: el rango de creación de gráfico horizontal
- Rng Y: el rango de creación de gráfico vertical
- Mrc X: espaciado entre marcas de graduación horizontal
- Mrc Y: espaciado entre marcas de graduación vertical

Los botones del menú en la primera página son:

- [[Editar]] abre un nuevo cuadro de edición para editar el valor que aparece en el campo seleccionado.
- [-Página 1/2 ▼-] muestra la segunda página de la configuración

Pulse [-Página 1/2 ▼-] para ver la segunda página de la configuración. Aquí, los campos son:

- Ejes: activa o desactiva los ejes
- Etiquetas: activa o desactiva las etiquetas del eje
- Puntos cuadr.: activa o desactiva los puntos de la cuadrícula
- Líneas cuadríc.: activa o desactiva las líneas de la cuadrícula
- Cursor: permite elegir entre cursores estándar, invertido y parpadeante
- Método: elegir el método utilizado para trazar el ajuste
- Conectar: conectar los puntos del gráfico de dispersión con el segmento (esto no es un ajuste)
- Ajuste: activa o desactiva el trazado de ajuste en la Vista de gráfico

Los botones del menú en la segunda página son:

- [[✓]] activa o desactiva la configuración actual
- [[Selec.]] permite elegir de un cuadro de elección
- [-▲ Página 2/2-] retorna a la primera página de la configuración

0301 Vista numérica de Var 2 estadística

ID 301 · título localizado en HelpEs

Presione Num para volver a esta vista en cualquier momento. Esta vista contiene una tabla con hasta diez columnas de datos, denominadas de C1 a C9 y C0.

Los botones del menú son:

- [[Editar]] abre una línea de entrada para editar el valor deseado
- [[Más]] abre un menú con opciones para editar la lista
- [[Ir a]] pasa a un elemento específico en la lista. Útil para listas muy grandes.
- [[Orden.]] organiza la columna actual en orden ascendente o descendente
- [[Crear]] genera una columna de datos basada en una fórmula algebraica
- [[Estado]] proporciona resúmenes de estadísticas de los análisis definidos actualmente (consulte Vista simbólica)

Introduzca sus datos de forma manual o almacene la lista de datos en C1, C2, etc. Utilice la función [[Make]] para crear los datos basados en una fórmula algebraica. También puede pegar los datos copiados de otra aplicación o de los Editores de lista y matriz.

También puede poner un nombre a cada columna de datos. Toque en el encabezado de la columna y luego edite el nombre o comience a escribir para introducir un nuevo nombre.

El menú [[Más]] contiene las siguientes opciones para editar una lista:

- Insertar Fila: inserta una nueva fila en la lista actual. La nueva fila contiene 0 como su elemento.
- Eliminar Columna: borra el contenido de la lista actual. Para eliminar un solo elemento, selecciónelo y pulse la tecla Retroceso.
- Seleccionar Fila: selecciona la fila actual. Una vez seleccionada, la fila se puede copiar. Columna: selecciona la lista actual. Una vez seleccionada, la lista se puede copiar. Cuadro: abre un cuadro de diálogo para seleccionar una matriz rectangular definida por un punto de partida y una ubicación final. También puede mantener pulsada una celda para iniciar la selección, y a continuación, arrastrar para seleccionar una matriz rectangular de elementos. Una vez seleccionada, la matriz se puede copiar. Intercambiar final: intercambia las celdas de inicio y final de la matriz de celdas rectangular seleccionada.
- Selección: activa y desactiva el modo de selección. También puede mantener pulsada una celda, y a continuación, arrastrar para seleccionar.
- Interc Columna: intercambia el contenido de dos columnas (listas).

1638 Crear columna

ID 1638 · título localizado en HelpEs

El asistente de datos de Crear columna es básicamente un método abreviado para usar el comando MAKELIST y almacenar los resultados en la Vista numérica. Los campos de este asistente son:

- Expresión: introduzca la expresión de generación para su lista de números reales
- Var: declare la variable independiente de su expresión. Todas las demás variables de su expresión se tomarán como constantes. Tenga en cuenta que su variable puede ser simulada; es decir, es posible que no aparezca en su expresión en absoluto.
- Inicio: ingrese el valor inicial para su variable
- Final: ingrese el valor final de su variable
- Paso: ingrese el valor de incremento para su variable
- Col: una lista desplegable para elegir el destino de su lista de números reales

Complete los campos y toque [[OK]] para generar la columna de números y guárdelos en la lista que especificó, o toque [[Cancelar]] para volver a la Vista numérica sin crear una lista. Use este asistente para crear con facilidad una lista de enteros aleatorios, una distribución de muestra, etc.

0302 Vista de estadísticas de Var 2 estadística

En la Vista numérica de la aplicación Var 2 estadísticas, pulse **[[Estado]]** para ver un resumen estadístico para todos los análisis activos que aparecen en la vista Estadísticas. De forma predeterminada, se muestran las estadísticas habituales de 2 variables y la tecla de menú **[[Estadísticas]]** tiene el punto blanco activo sobre ella. Las teclas de menú en esta vista son:

- **[[Más]]** abre un menú que le permite seleccionar y copiar varias celdas. A continuación, puede pegar el contenido de estas celdas en otros lugares.
- **[[Estadísticas]]** muestra un resumen de las estadísticas de 2 variables comunes, como el coeficiente de correlación y las covarianzas de muestra y población.
- **[[X]]** muestra un resumen de las estadísticas de la variable independiente, como la media de x y su desviación estándar.
- **[[Y]]** muestra un resumen de las estadísticas de la variable dependiente, como la media de Y y su desviación estándar.
- **[[OK]]** vuelve a la Vista numérica

Utilice el menú **[[Más]]** para seleccionar una o más de las estadísticas en una página y luego pulse Shift View (Copy) para copiar la matriz en el portapapeles. A continuación, puede pulsar Shift Menu (Paste) para abrir el portapapeles y pegar la matriz en cualquier parte del sistema, como los Editores de lista o matriz o la Vista numérica de la aplicación Hoja de cálculo.

Cuando haya terminado de revisar el resumen de las estadísticas, pulse **[[OK]]** para volver a la Vista numérica.

0303 Variables de Var 2 estadística

Para mostrar las variables relacionadas con la Aplicación Var 2 estadística, presione Vars, toque **[[Apl.]]** y seleccione Var 2 estadística.

La Aplicación Var 2 estadística cuenta con las siguientes variables:

- Resultados (consulte a continuación)
- Simbólica (consulte a continuación)
- Gráfico (consulte Variables de aplicaciones comunes más arriba)
- Numérica (consulte a continuación)
- Modos (consulte Variables de aplicaciones comunes más arriba)

0304 Variables de resultados

Las variables de la aplicación Var 2 estadística guardan los resultados de los cálculos efectuados al pulsar el botón **[[Estado]]** en la Vista numérica de la aplicación o al ejecutar el comando Do2VarStats.

0305 Variable de aplicación Nbltem

Nbltem contiene el número de puntos de datos en el análisis de 2 variables actual (S1-S5).

0306 Variable de aplicación Corr

Corr contiene el coeficiente de correlación del último cálculo de resúmenes de estadísticas.

0307 Variable de la aplicación CoefDet

CoefDet contiene el coeficiente de determinación a partir del último cálculo de las estadísticas resumidas. Este valor se basa en el tipo de ajuste elegido.

0308 Variable de aplicación sCov

sCov contiene la covarianza de muestra del análisis estadístico actual de dos variables (S1-S5).

0309 Variable de aplicación σ_{Cov}

ID 309 · título recuperado de la página española

σ_{Cov} contiene la covarianza de la población del análisis estadístico de 2 variables actual (S1-S5).

0310 Variable de aplicación ΣXY

ID 310 · título recuperado de la página española

ΣXY contiene la suma de los productos X-Y para el análisis estadístico de 2 variables actual (S1-S5).

0311 Variable de aplicación MeanX

ID 311 · título recuperado de la página española

MeanX contiene el promedio de los valores independientes (X) del análisis estadístico de 2 variables actual (S1-S5).

0312 Variable de aplicación ΣX

ID 312 · título recuperado de la página española

ΣX contiene la suma de los valores independientes (X) del análisis estadístico de 2 variables actual (S1-S5).

0313 Variable de aplicación ΣX^2

ID 313 · título recuperado de la página española

ΣX^2 contiene la suma de los cuadrados de los valores independientes (X) del análisis estadísticos de 2 variables actual (S1-S5).

0314 Variable de aplicación sX

ID 314 · título recuperado de la página española

sX contiene la desviación estándar de muestra de los valores independientes (X) del análisis estadístico de 2 variables actual (S1-S5).

0315 Variable de aplicación σX

ID 315 · título recuperado de la página española

σX contiene la desviación estándar de la población de los valores independientes (X) del análisis estadístico actual de 2 variables (S1-S5).

0316 Variable de aplicación serrX

ID 316 · título recuperado de la página española

serrX contiene el error estándar de los valores independientes (X) del análisis estadístico actual de dos variables (S1-S5).

0317 Variable de aplicación ssX

ID 317 · título recuperado de la página española

ssX contiene la suma de las desviaciones al cuadrado de x a partir de la media de x de los valores independientes (X) del análisis estadístico de 2 variables (S1-S5) actual.

0318 Variable de aplicación MeanY

ID 318 · título recuperado de la página española

MeanY contiene el promedio de los valores dependientes (Y) del análisis estadístico de 2 variables actual (S1-S5).

0319 Variable de la aplicación ΣY

ID 319 · título recuperado de la página española

ΣY contiene la suma de los valores dependientes (Y) del análisis estadístico actual de 2 variables (S1-S5).

0320 Variable de aplicación ΣY

ID 320 · título recuperado de la página española

ΣY contiene la suma de los cuadrados de los valores dependientes (Y) del análisis estadístico actual de 2 variables (S1-S5).

0321 Variable de aplicación sY

ID 321 · título recuperado de la página española

sY contiene la desviación estándar de muestra de los valores dependientes (Y) del análisis estadístico de 2 variables actual (S1-S5).

0322 Variable de la aplicación σY

ID 322 · título recuperado de la página española

σY contiene la desviación estándar de la población de los valores dependientes (Y) del análisis estadístico actual de 2 variables (S1-S5).

0323 Variable de aplicación $serrY$

ID 323 · título recuperado de la página española

$serrY$ contiene el error estándar de los valores dependientes (Y) del análisis estadístico actual de dos variables (S1-S5).

0324 Variable de aplicación ssY

ID 324 · título recuperado de la página española

ssY contiene la suma de las desviaciones al cuadrado de y a partir de la media de y de los valores independientes (Y) del análisis estadístico de 2 variables (S1-S5) actual.

0325 Variables simbólicas

ID 325 · título localizado en HelpEs

Las variables de la aplicación Var 2 estadística son S1-S5. Estas variables contienen los datos que definen un análisis estadístico de 2 variables. S1 devuelve una lista de los objetos que definen S1. Cada lista contiene los siguientes elementos, en orden:

- Una cadena o expresión (en comillas simples) que define la lista de datos de la variable independiente
- Una cadena o expresión (en comillas simples) que define la lista de datos de la variable dependiente
- Una cadena o expresión que define opcionalmente las frecuencias de la lista de datos dependientes
- El número de tipo de ajuste (ver más abajo)
- La expresión de ajuste
- El color del gráfico de dispersión
- El número de tipo de marcas del punto del gráfico de dispersión
- El color del gráfico de ajuste

El número de tipo de ajuste es un número entero del 1-13 que controla qué tipo de gráfico estadístico se utiliza con cada una de las variables S1- S5. La correspondencia se muestra a continuación.

- 1 Lineal
- 2 Logarítmica
- 3 Exponencial
- 4 Potencia
- 5 Exponente
- 6 Inversa
- 7 Logística
- 8 Cuadrática
- 9 Cúbica
- 10 Cuártica
- 11 Trigonométrica
- 12 Línea mediana-mediana
- 11 Definida por el usuario

El número de tipo de marca del punto del gráfico de dispersión es un número entero de 1 a 9 que controla qué gráfico se utiliza para representar cada punto en un gráfico de dispersión. La correspondencia se muestra a continuación.

- 1 punto pequeño hueco
- 2 cuadrado pequeño hueco
- 3 x fina
- 4 cruz hueca
- 5 diamante pequeño hueco
- 6 x gruesa
- 7 punto pequeño sólido
- 8 diamante fino
- 9 punto grande hueco

Ejemplo

S1:={"C1", "C2", "", 1, "", #FF:24h, 1, #FF:24h} establece C1 como los datos independientes, C2 como los datos dependientes, no hay frecuencias para los datos dependientes, un ajuste lineal, no hay ecuación específica para ese ajuste lineal aún, un gráfico de dispersión de color azul con tipo de marca 1, y un gráfico de ajuste azul.

0326 Variables numéricas

ID 326 · título localizado en HelpEs

Las variables de la aplicación numérica Var 2 estadística son de C1 a C9 y C0. Cada una representa un único conjunto de datos y contiene los valores de ese conjunto de datos. Todas son variables de lista y son compatibles con las variables de lista de Var 1 estadística D0-D9, así como las variables de Inicio L0-L9.

0327 Funciones de aplicación Var 2 estadística

ID 327 · título localizado en HelpEs

La aplicación Var 2 estadística tiene una serie de funciones. Algunas se han diseñado para calcular resúmenes de estadísticas basados en uno de los análisis estadísticos (S1-S5) definidos en la Vista simbólica de la aplicación Var 2 estadística. Otras predicen valores de X e Y según el ajuste especificado en uno de los análisis.

0328 Variable de aplicación PredX

ID 328 · título recuperado de la página española

Sintaxis

PredX(Valor_Y)

Predecir X. Utiliza el ajuste del primer análisis activo (S1-S5) hallado para predecir el valor de x para el valor de Y dado.

0329 Variable de aplicación PredY

ID 329 · título recuperado de la página española

Sintaxis

PredY(Valor_X)

Predecir Y. Utiliza el ajuste del primer análisis activo (S1-S5) hallado para predecir el valor de y para el valor de x dado.

0330 Función de aplicación Resid

ID 330 · título recuperado de la página española

Sintaxis

Resid(Sn)
Resid()

Residuales. Calcula una lista de residuales en función de la base de datos de columna y un ajuste definido en la Vista simbólica a través de S1-S5.

Ejemplos

[Ejemplo gráfico E1] busca el primer análisis definido activo en la Vista simbólica (S1-S5).
[Ejemplo gráfico E2] utiliza análisis S3

0331 Función de la aplicación Do2VStats

ID 331 · título recuperado de la página española

Sintaxis

Do2VStats(Sn)

Efectúa los mismos cálculos que cuando se toca [[Estado]] en la Vista numérica de la aplicación Var 2 estadística, y guarda los resultados en las variables de resultados de Var 2 estadística correspondientes. Sn debe corresponder a una de las variables S1-S5 de la Vista simbólica de la aplicación Var 2 estadística.

0332 Función de aplicación SetDepend

ID 332 · título recuperado de la página española

Sintaxis

SetDepend(Sn, Cn)

Establecer columna dependiente. Establece la columna dependiente para uno de los análisis estadísticos S1-S5 en una de las variables de columna C0-C9.

Ejemplos

[Ejemplo gráfico E1] configura la columna dependiente para el análisis S1 en la columna C3.

0333 Función de aplicación SetIndep

ID 333 · título recuperado de la página española

Sintaxis

SetIndep(Sn, Cn)

Establecer columna independiente. Establece la columna independiente para uno de los análisis estadísticos S1-S5 en una de las variables de columna C0-C9.

Ejemplos

[Ejemplo gráfico E1] configura la columna independiente para el análisis S1 en la columna C2.

0294 Función de aplicación CHECK

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función de aplicación CHECK.](#)

0295 Función de aplicación UNCHECK

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función de aplicación UNCHECK.](#)

0296 Función de aplicación ISCHECK

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función de aplicación ISCHECK.](#)

0334 Aplicación Inferencia

ID 334 · título localizado en HelpEs

La aplicación Inferencia contiene herramientas de estadística inferencial, incluida la creación de intervalos de confianza y pruebas de hipótesis. Esta aplicación puede importar estadísticos resumidos de cualquier columna de las aplicaciones Var 1 estadística o Var 2 estadística. Los intervalos de confianza y las pruebas de hipótesis se basan en la distribución normal Z o en la distribución t de Student. Los resultados se pueden mostrar de forma numérica y gráfica.

Para iniciar la aplicación Inferencia, vaya a la biblioteca de aplicaciones y toque el icono de la aplicación Inferencia. También puede utilizar la rueda de selección para seleccionar el icono de la aplicación Inferencia y, a continuación, toque **[[Inicio]]** o presione Enter para iniciar la aplicación.

Aunque los intervalos y las pruebas de esta aplicación se limitan a las distribuciones normal y t de Student, HP Prime dispone de un conjunto completo de funciones de densidad de probabilidad. Presione la tecla Caja de herramientas, toque **[[Matem.]]** y seleccione Probabilidad. Verá las categorías Densidad, Acumulada e Inversa (acumulada). En cada una encontrará Normal, T, Chi cuadrado, F, Binomial y Poisson.

0335 Vista simbólica de Inferencia

ID 335 · título localizado en HelpEs

La Vista simbólica de Inferencia contiene la configuración para definir un intervalo de confianza o una prueba de hipótesis. Esta visión también le permite seleccionar una prueba Chi cuadrado o realizar inferencia de regresión lineal.

Los campos son los siguientes:

- Método: elige entre prueba de hipótesis, intervalos de confianza, Chi cuadrado, inferencia para regresión o ANOVA
- Tipo: elige un cálculo específico dentro del método seleccionado, donde esté disponible
- Hipót. Alt: elige una de las tres hipótesis alternativas (solo para pruebas de hipótesis).
- Esperada: elige entre ingresar probabilidades esperadas y conteos esperados para la prueba de bondad de ajuste (GoF) de Chi cuadrado

El único botón del menú es el siguiente:

- **[[Selec.]]** permite elegir de un cuadro de elección

Una vez que haya seleccionado un cálculo específico, pulse la tecla Num para ir a la Vista numérica e introduzca los datos para el cálculo.

0336 Vista de gráfico de Inferencia

ID 336 · título localizado en HelpEs

La Vista de gráfico de Inferencia muestra gráficamente los resultados que usted ve cuando toca **[[Calc]]** desde la Vista numérica. No todos los cálculos incluyen una Vista de gráfico.

0337 Configuración del gráfico de inferencia

ID 337 · título localizado en HelpEs

Presione Shift Plot para entrar a la Configuración del gráfico de inferencia. La página 1 de la Configuración del gráfico contiene parámetros que controlan la apariencia de los gráficos de inferencia.

Los campos son los siguientes:

- Anch H: ancho del intervalo para histogramas
- Rng H: rango de datos que desea trazar.
- Rng X: el rango horizontal de la ventana de gráfico
- Rng Y: el rango vertical de la ventana de gráfico
- Mrc X: espaciado entre marcas de graduación horizontal
- Mrc Y: espaciado entre marcas de graduación vertical

0338 Vista numérica de Inferencia

ID 338 · título localizado en HelpEs

La Vista numérica de la aplicación Inferencia está diseñada para que introduzca los datos necesarios para el cálculo seleccionado en la Vista simbólica.

Pruebas de hipótesis e intervalos de confianza

Para pruebas de hipótesis e intervalos de confianza, esta vista contiene campos para las estadísticas de muestra (por ejemplo, muestras promedio y tamaño de la muestra), los parámetros de la población (por ejemplo, hipótesis nula promedio y la desviación estándar, σ) y el nivel de significancia.

Para pruebas de hipótesis e intervalos de confianza, los botones de menú son los siguientes::

- **[[Editar]]** abre una línea de entrada para editar el valor deseado
- **[[Impor.]]** permite importar estadísticas (por ejemplo n, promedio, etc.) desde cualquier columna de la aplicación Var 1 estadística o Var 2 estadística (o cualquier aplicación basada en estas dos)
- **[[Calc]]** permite calcular los resultados y los muestra numéricamente en una tabla

Introduzca los valores de la prueba o del intervalo y, a continuación, toque **[[Calc]]** para ver los resultados.

Pruebas de Chi cuadrado

Para la prueba de GOF de chi cuadrado, la Vista numérica presenta dos listas. La primera lista, ObsList, es para los conteos observados. La segunda lista es ProbList para las probabilidades esperadas o ExpList para los conteos esperados, dependiendo de la opción que seleccionó en la Vista simbólica para el campo Esperado. Introduzca los conteos observados en la primera lista y las probabilidades o los conteos esperados en la segunda lista. Toque **[[Calc]]** para ver los resultados.

Los botones del menú son:

- **[[Editar]]** abre una línea de entrada para editar el valor deseado
- **[[Más]]** abre un menú con opciones para editar la lista
- **[[Ir a]]** pasa a un elemento específico en la lista. Útil para listas muy grandes.
- **[[Crear]]** genera una columna de datos basada en una fórmula algebraica
- **[[Calc]]** muestra los resultados de las pruebas

Para la prueba Chi-cuadrado de 2 vías, la Vista numérica presenta una matriz llamada ObsMat, en la que se ingresa la matriz de conteos observados. Cuando haya terminado, pulse **[[Calc]]** para ver los resultados. La parte superior de la pantalla mostrará el valor estadístico de Chi cuadrado, la probabilidad y los grados de libertad.

Los botones de menú son los mismos que para la prueba GoF, excepto que no hay ninguna opción **[[Crear]]**. También hay una tecla de menú **[[Ir]]** que alterna entre mover el cursor hacia la derecha, hacia abajo o no moverlo después de presionar Enter.

Inferencia para regresión lineal

Para todas las opciones de regresión lineal, la Vista numérica presenta dos listas; Xlist para los valores de x (la variable explicativa) y Ylist para los valores de y (la variable de respuesta). Introduzca sus datos en estas dos variables. Presione la tecla Plot para ver un diagrama de dispersión de sus datos. Utilice la rueda hacia arriba/abajo para ver un diagrama de dispersión de los residuos y un gráfico de probabilidad normal de los residuos. Estas gráficas le ayudarán a evaluar si sus datos cumplen o no los criterios para la prueba t lineal, etc. Cuando esté listo, pulse **[[Calc]]**. Se abrirá un asistente que lo guiará por el resto de los pasos, por ejemplo, cómo introducir un nivel de confianza o un valor x, según el cálculo que haya elegido en la Vista simbólica.

Los botones de menú aquí son los mismos que para la prueba GoF Chi-cuadrado en la Vista numérica.

ANOVA

Para el análisis de varianza, la Vista numérica presenta un conjunto de columnas. Introduzca cada conjunto de datos involucrados en el análisis en su propia columna y después pulse **[[Calc]]**. Se mostrarán los resultados del análisis de varianza.

Los botones de menú aquí son los mismos que para la prueba GoF Chi-cuadrado en la Vista numérica.

0339 Vista Calc de Inferencia

ID 339 · título localizado en HelpEs

Pulse **[[Calc]]** para ver los resultados de cálculo. Aparece la pantalla con los resultados de Inferencia. Esta vista varía dependiendo de los resultados de cálculo específico que se muestran.

En general, los resultados se muestran en una tabla siempre que sea posible. Cada valor se etiqueta y se proporciona texto de ayuda para describir cada valor.

Pruebas de hipótesis

Para las pruebas de hipótesis, **[[Calc]]** muestra el resultado de la prueba, los valores de la prueba, la probabilidad y los valores críticos. Las teclas de menú aquí incluyen el menú **[[Más]]** para ayudarle a copiar los resultados en otro lugar. Toque la tecla de menú **[[OK]]** para volver a la Vista numérica cuando haya terminado.

Intervalos de confianza

Para los intervalos de confianza, **[[Calc]]** muestra el valor de C, los grados de libertad, el valor crítico y los límites inferior y superior del intervalo de confianza. Las teclas de menú aquí son las mismas de las Pruebas de hipótesis.

Prueba de bondad de ajuste de chi cuadrado

Para la prueba de bondad de ajuste, **[[Calc]]** muestra el valor de chi cuadrado y su probabilidad, junto con los

grados de libertad. Las teclas del menú son:

- [[Más]] abre el menú Más
- [[Estads]] muestra los resultados predeterminados descritos anteriormente
- [[Exp]] cambia para mostrar los recuentos esperados
- [[Cont]] cambia para mostrar las contribuciones de chi cuadrado por categoría
- [[OK]] vuelve a la Vista numérica

Prueba de Chi cuadrado de 2 vías

Para la prueba de dos vías, [[Calc]] muestra el valor de chi cuadrado y su probabilidad, junto con los grados de libertad. Las teclas del menú son las mismas que para la prueba de bondad de ajuste.

ANOVA

Para el análisis de 1 vía de la variancia, [[Calc]] muestra los grados de libertad, el valor de distribución de f, etc. Las teclas de menú son [[Más]] (para seleccionar varias celdas con el fin de copiarlas y pegarlas en otro lugar) y [[OK]] (para volver a la Vista numérica).

Consulte las páginas de ayuda individual para:

- Pruebas de hipótesis
- Intervalos de confianza
- Prueba de bondad de ajuste Chi cuadrado
- Prueba de Chi cuadrado de 2 vías
- Prueba T de regresión lineal
- Intervalo de confianza para la pendiente
- Intervalo de confianza para interceptancia
- Intervalo de confianza para la respuesta promedio
- Intervalo de predicción para la respuesta futura
- ANOVA

0340 Resultados de la prueba de hipótesis

ID 340 · título localizado en HelpEs

Una vez que toca [[Calc]] desde la Vista numérica, los resultados de la prueba de hipótesis se muestran en una tabla. Por lo general, estos resultados incluyen los siguientes valores:

- Resultado: 0 para rechazar o 1 para no rechazar la hipótesis nula
- Valor de prueba: el valor Z o t calculado para la prueba
- Valor estadístico de prueba: el valor de la estadística bajo escrutinio asociado con el valor de prueba
- P: la probabilidad
- Valor de prueba crítica: valores límites de la prueba para su prueba
- Valor crítico estadístico: los valores límite para la estadística bajo escrutinio

Los botones del menú son:

- [[Más]] abre el menú Más, con opciones para seleccionar varias celdas para copiar y luego pegar en otra parte
- [[OK]] permite regresar a la Vista numérica de la aplicación

Utilice la rueda o pulse para moverse por la tabla. Pulse la tecla de menú [[Más]] para abrir un menú de opciones que lo ayudará a seleccionar varias celdas para copiar y pegar en otro lugar. También puede mantener pulsada una celda, luego arrastrarla para seleccionar una matriz rectangular de celdas para copiar y pegar en otro lugar.

Pulse la tecla de menú [[OK]] para volver a la Vista numérica. Ahora puede pulsar la tecla Plot para ver los resultados de forma gráfica también.

0341 Resultados de intervalos de confianza

ID 341 · título localizado en HelpEs

Una vez que toca [[Calc]] en la Vista numérica, los resultados del cálculo de intervalo de confianza se muestran en una tabla. Por lo general, estos resultados incluyen los siguientes valores:

- C: el nivel de confianza que ingresó
- Valores de prueba crítica: los valores límite de la prueba asociados a su nivel de confianza
- Inferior: el límite inferior del intervalo de confianza

- Superior: el límite superior del intervalo de confianza

Los botones del menú son:

- [[Más]] abre el menú Más, con opciones para seleccionar varias celdas para copiar y luego pegar en otra parte
- [[OK]] permite regresar a la Vista numérica de la aplicación

Utilice la rueda o pulse para moverse por la tabla. Pulse la tecla de menú [[Más]] para abrir un menú de opciones que lo ayudará a seleccionar varias celdas para copiar y pegar en otro lugar. También puede mantener pulsada una celda, luego arrastrarla para seleccionar una matriz rectangular de celdas para copiar y pegar en otro lugar.

Pulse la tecla de menú [[OK]] para volver a la Vista numérica. Ahora puede pulsar la tecla Plot para ver los resultados de forma gráfica también.

0342 Resultados GoF de Chi cuadrado

ID 342 · título localizado en HelpEs

Una vez que toca [[Calc]] desde la Vista numérica, se muestran los resultados del cálculo de bondad de ajuste Chi-cuadrado en una tabla que contiene los siguientes valores:

- χ^2 : el valor estadístico Chi cuadrado
- P: la probabilidad
- DF: los grados de libertad

Las teclas de menú son:

- [[Más]] abre el menú Más, con opciones para seleccionar varias celdas para copiar y luego pegar en otra parte
- [[Estads.]] toque para mostrar los resultados de la prueba (en forma predeterminada)
- [[Esp]] toque para mostrar la lista de los conteos esperados
- [[Cont]] toque para mostrar la lista de contribuciones de Chi cuadrado
- [[OK]] toque para regresar a la Vista numérica

Utilice la rueda o pulse para desplazarse por la tabla. Utilice el menú [[Más]] para seleccionar varias celdas para copiar y pegar en otro lugar. También puede mantener presionada una celda y arrastrarla para seleccionar una matriz rectangular de celdas.

Pulse la tecla de menú [[OK]] para volver a la Vista numérica.

0343 Resultados de la prueba Chi cuadrado de 2 vías

ID 343 · título localizado en HelpEs

Una vez toca [[Calc]] en la Vista numérica, los resultados de su prueba Chi-cuadrado de 2 vías se muestran en una tabla que contiene los siguientes valores:

- χ^2 : el valor estadístico Chi cuadrado
- P: la probabilidad
- DF: los grados de libertad

Las teclas de menú son:

- [[Más]] abre el menú Más, con opciones para seleccionar varias celdas para copiar y luego pegar en otra parte
- [[Esp]] toque para mostrar la matriz de los conteos esperados (pulse [[OK]] para salir)
- [[Cont]] toque para visualizar la matriz de las contribuciones de Chi cuadrado (pulse [[OK]] para salir)
- [[OK]] toque para regresar a la Vista numérica

Utilice la rueda o pulse para desplazarse por la tabla. Utilice el menú [[Más]] para seleccionar varias celdas para copiar y pegar en otro lugar. También puede mantener presionada una celda y arrastrarla para seleccionar una matriz rectangular de celdas.

Pulse la tecla de menú [[OK]] para volver a la Vista numérica.

0344 Resultados de la prueba T de la regresión lineal

ID 344 · título localizado en HelpEs

Una vez que toca [[Calc]] en la Vista numérica, los resultados de la prueba t de regresión lineal se muestran en una tabla que contiene los siguientes valores:

- Prueba T: el valor calculado de la prueba t
- P: la probabilidad

- DF: los grados de libertad
- β_0 : la regresión lineal de la ecuación de interceptación
- β_1 : la regresión lineal de la ecuación de pendiente
- serrLine: el error estándar de la línea
- serrSlope: el error estándar de la pendiente
- serrInt: el error estándar de la interceptación
- r: el valor de correlación
- R^2 : el coeficiente de determinación

Utilice la rueda o pulse para desplazarse por la tabla. Pulse la tecla de menú **[[Más]]** para abrir un menú de opciones que lo ayudará a seleccionar varias celdas para copiar y pegar en otro lugar. También puede mantener pulsada una celda, luego arrastrarla para seleccionar una matriz rectangular de celdas para copiar y pegar en otro lugar.

Pulse la tecla de menú **[[OK]]** para volver a la Vista numérica.

0345 CI para resultados de pendiente

ID 345 · título localizado en HelpEs

Una vez que toca **[[Calc]]** desde la Vista numérica, los resultados del cálculo del intervalo de confianza se muestran en una tabla que contiene los valores siguientes:

- C: el nivel de confianza que usted ingresó en el asistente
- T Crítico: el valor crítico de t asociado a su nivel de confianza
- DF: los grados de libertad
- β_1 : la regresión lineal de la ecuación de pendiente
- serrLine: el error estándar de la línea
- Inferior: el límite inferior del intervalo de confianza para la pendiente
- Superior: el límite superior del intervalo de confianza para la pendiente

Utilice la rueda o pulse para moverse por la tabla. Pulse la tecla de menú **[[Más]]** para abrir un menú de opciones que lo ayudará a seleccionar varias celdas para copiar y pegar en otro lugar. También puede mantener pulsada una celda, luego arrastrarla para seleccionar una matriz rectangular de celdas para copiar y pegar en otro lugar.

Pulse la tecla de menú **[[OK]]** para volver a la Vista numérica.

0346 CI para resultados de Interceptar

ID 346 · título localizado en HelpEs

Una vez que toca **[[Calc]]** desde la Vista numérica, los resultados del cálculo del intervalo de confianza se muestran en una tabla que contiene los valores siguientes:

- C: el nivel de confianza que usted ingresó en el asistente
- T Crítico: el valor crítico para prueba t asociado a su nivel de confianza
- DF: los grados de libertad
- β_0 : la regresión lineal de la ecuación de interceptación
- serrLine: el error estándar de la línea
- Inferior: el límite inferior del intervalo de confianza para la interceptación
- Superior: el límite superior del intervalo de confianza para la interceptación

Utilice la rueda o pulse para moverse por la tabla. Pulse la tecla de menú **[[Más]]** para abrir un menú de opciones que lo ayudará a seleccionar varias celdas para copiar y pegar en otro lugar. También puede mantener pulsada una celda, luego arrastrarla para seleccionar una matriz rectangular de celdas para copiar y pegar en otro lugar.

Pulse la tecla de menú **[[OK]]** para volver a la Vista numérica.

0347 CI para resultados de respuesta promedio

ID 347 · título localizado en HelpEs

Una vez que toca **[[Calc]]** desde la Vista numérica, los resultados del cálculo del intervalo de confianza se muestran en una tabla que contiene los valores siguientes:

- C: el nivel de confianza que usted ingresó en el asistente

- $\hat{y}$: el valor de respuesta promedio para el valor de x que especificó en el asistente
- DF: los grados de libertad
- $serr \hat{y}$: el error estándar de la respuesta promedio
- Inferior: el límite inferior del intervalo de confianza para la respuesta promedio
- Superior: el límite superior del intervalo de confianza para la respuesta promedio

Utilice la rueda o pulse para moverse por la tabla. Pulse la tecla de menú **[[Más]]** para abrir un menú de opciones que lo ayudará a seleccionar varias celdas para copiar y pegar en otro lugar. También puede mantener pulsada una celda, luego arrastrarla para seleccionar una matriz rectangular de celdas para copiar y pegar en otro lugar.

Pulse la tecla de menú **[[OK]]** para volver a la Vista numérica.

0348 Resultados de intervalo de predicción

ID 348 · título localizado en HelpEs

Una vez que toca **[[Calc]]** desde la Vista numérica, los resultados del cálculo del intervalo de predicción se muestran en una tabla que contiene los valores siguientes:

- C: el nivel de confianza que usted ingresó en el asistente
- $\hat{y}$: el valor de respuesta promedio para el valor de x futuro que especificó en el asistente
- DF: los grados de libertad
- $serr \hat{y}$: el error estándar de la respuesta promedio
- Inferior: el límite inferior del intervalo de confianza para la respuesta promedio
- Superior: el límite superior del intervalo de confianza para la respuesta promedio

Utilice la rueda o pulse para moverse por la tabla. Pulse la tecla de menú **[[Más]]** para abrir un menú de opciones que lo ayudará a seleccionar varias celdas para copiar y pegar en otro lugar. También puede mantener pulsada una celda, luego arrastrarla para seleccionar una matriz rectangular de celdas para copiar y pegar en otro lugar.

Pulse la tecla de menú **[[OK]]** para volver a la Vista numérica.

0349 Resultados ANOVA

ID 349 · título localizado en HelpEs

Una vez que toca **[[Calc]]** en la Vista numérica, los resultados de su análisis de varianza de 1 vía se muestran en una tabla que contiene los siguientes valores:

- F: el valor F
- P: la probabilidad asociada con el valor F
- DF: los grados de libertad de los tratamientos
- SS: la suma de las cuadráticas de los tratamientos
- MS: la cuadrática media de los tratamientos
- Dferr: los grados de libertad de los errores
- SSerr: la suma de las cuadráticas de los errores
- Mserr: la cuadrática media de los errores

Utilice la rueda o pulse para moverse por la tabla. Pulse la tecla de menú **[[Más]]** para abrir un menú de opciones que lo ayudará a seleccionar varias celdas para copiar y pegar en otro lugar. También puede mantener pulsada una celda, luego arrastrarla para seleccionar una matriz rectangular de celdas para copiar y pegar en otro lugar.

Pulse la tecla de menú **[[OK]]** para volver a la Vista numérica.

0350 Variables de la aplicación Inferencia

ID 350 · título localizado en HelpEs

Además de las variables comunes de aplicación, la aplicación Inferencia posee las variables de aplicación Simbólica, Numérica y Resultados. Cada una de estas categorías contiene las variables de aplicación utilizadas en la vista correspondiente de la aplicación.

0351 Variables de resultados

ID 351 · título localizado en HelpEs

Las variables de los resultados de la aplicación Inferencia guardan los cálculos realizados cuando se toca el botón de menú [[Calc]] en la Vista numérica de Inferencia o cuando se ejecuta el comando DoInference.

0352 Variable de aplicación Result

ID 352 · título recuperado de la página española

Para las pruebas de hipótesis, contiene 0 o 1, para indicar el rechazo o fracaso para rechazar la hipótesis nula.

0353 Variable de aplicación TestScore

ID 353 · título recuperado de la página española

TestScore contiene el valor de distribución de Z- o t- calculado desde la prueba de hipótesis o de entradas de intervalos de confianza.

0354 Variable de aplicación TestValue

ID 354 · título recuperado de la página española

TestValue contiene el valor de la variable experimental asociada con el valor actual en TestScore de la variable de la aplicación.

0355 Variable de aplicación CritScore

ID 355 · título recuperado de la página española

CritScore contiene el valor de la distribución de Z o de t asociada con el valor α de la entrada

0356 Variable de aplicación CritVal1

ID 356 · título recuperado de la página española

CritVal1 contiene el valor inferior crítico de la variable experimental asociada con el valor negativo de TestScore que se calculó a partir del nivel α de la entrada.

0357 Variable de aplicación CritVal2

ID 357 · título recuperado de la página española

CritVal2 contiene el valor superior crítico de la variable experimental asociada con el valor positivo de TestScore que se calculó a partir del nivel α de la entrada.

0358 Variable de aplicación Prob

ID 358 · título recuperado de la página española

Prob contiene la probabilidad asociada con el valor TestScore.

0359 Variable de aplicación DF

ID 359 · título recuperado de la página española

DF contiene los grados de libertad de las pruebas de t.

0360 Variable de aplicación ContribList

ID 360 · título recuperado de la página española

ContribList es una lista que contiene las contribuciones de Chi cuadrado para la última prueba de Chi cuadrado de bondad de ajuste.

0398 Variable de la aplicación ExpMat

ID 398 · título recuperado de la página española

ExpMat es una matriz que contiene los datos de conteo esperados del última cálculo de prueba de Chi cuadrado de 2 vías.

0361 Variable de aplicación ContribMat

ID 361 · título recuperado de la página española

ContribMat es una matriz que contiene las contribuciones de Chi cuadrado para la última prueba de Chi cuadrado de 2 vías.

0362 Variable de aplicaciónSlope

ID 362 · título recuperado de la página española

Slope contiene el valor de la pendiente de la última prueba t de regresión lineal.

0363 Variable de aplicación Inter

ID 363 · título recuperado de la página española

Inter contiene el valor de la interceptación de la última prueba t de regresión lineal.

0364 Variable de aplicación corr

ID 364 · título recuperado de la página española

corr contiene el valor de la correlación de la última prueba T de regresión lineal.

0365 Variable de aplicacióncoefDet

ID 365 · título recuperado de la página española

coefDet contiene el valor del coeficiente de determinación a partir de la última prueba t de regresión lineal.

0366 Variable de aplicaciónserrLine

ID 366 · título recuperado de la página española

serrLine contiene el valor del error estándar de la línea de la última prueba T de regresión lineal.

0367 Variable de aplicación serrSlope

ID 367 · título recuperado de la página española

serrSlope contiene el valor del error estándar de la pendiente de la última prueba t de regresión lineal o intervalo de confianza de la pendiente.

0368 Variable de aplicación serrInter

ID 368 · título recuperado de la página española

serrInter contiene el valor del error estándar de la interceptación de la última prueba T de regresión lineal o intervalo de confianza para la interceptación.

0369 Variable de aplicaciónYval

ID 369 · título recuperado de la página española

Yval contiene el valor de $\hat{y}$ desde el último intervalo de predicción o el cálculo del intervalo de respuesta promedio.

0370 Variable de aplicaciónserrY

ID 370 · título recuperado de la página española

serrY contiene el valor del error estándar de $\hat{y}$ desde el último intervalo de predicción o cálculo de intervalo de respuesta promedio.

0371 Variable de aplicación Xval

ID 371 · título recuperado de la página española

Xval contiene el valor de la variable explicativa (X) a partir del último intervalo de respuesta promedio o cálculo de intervalo de predicción.

0372 Var de apl SS

ID 372 · título recuperado de la página española

SS contiene el valor de la suma de las cuadráticas de los tratamientos del último cálculo ANOVA de 1 vía.

0373 Var de apl SSerr

ID 373 · título recuperado de la página española

SSerr contiene el valor de la suma de las cuadráticas de los errores del último cálculo ANOVA de 1 vía.

0374 Var de apl MS

ID 374 · título recuperado de la página española

MS contiene el valor de las cuadráticas medias para los tratamientos del último cálculo ANOVA de 1 vía.

0375 Var de apl MSerr

ID 375 · título recuperado de la página española

MSerr contiene el valor de las cuadráticas medias para los errores del último cálculo ANOVA de 1 vía.

0376 Variable de la aplicación Fval

ID 376 · título recuperado de la página española

Fval contiene el valor del estadístico F del último cálculo ANOVA de 1 vía.

0377 Var de apl DFerr

ID 377 · título recuperado de la página española

DFerr contiene el valor de los grados de libertad de los errores del último cálculo ANOVA de 1 vía.

0378 Variables numéricas

ID 378 · título localizado en HelpEs

Variables numéricas de la aplicación Inferencia corresponden a los campos en la Vista numérica para las diversas pruebas e intervalos de confianza.

0379 Variable de la aplicación Alpha

ID 379 · título recuperado de la página española

Establece el nivel alpha para la prueba de hipótesis.

Alpha:=n, donde $0 < n < 1$, establece el nivel alfa en n.

0380 Variable de aplicación Conf

ID 380 · título recuperado de la página española

Establece el nivel de confianza para el intervalo de confianza.

Conf:=n, donde $0 < n < 1$, establece el nivel de confianza en n.

0381 Variable de aplicación Mean₁

ID 381 · título recuperado de la página española

Establece el valor del promedio de una muestra para una prueba de hipótesis o intervalo de confianza de un (1) promedio. Para una prueba o intervalo de dos promedios, establece el valor del promedio de la primera muestra.

n ► Mean₁ establece el valor de Mean₁ para n.

0382 Variable de aplicación Mean₂

ID 382 · título recuperado de la página española

Para una prueba o intervalo de dos promedios, establece el valor del promedio de la segunda muestra.

n ► Mean₂ establece el valor de Mean₂ para n.

0383 Variable de aplicación σ_1

ID 383 · título recuperado de la página española

Configura la desviación estándar de la población para una prueba de hipótesis o un intervalo de confianza que impliquen 1 o 2 promedios y una distribución normal. Para una prueba o intervalo que implique la diferencia de dos promedios, configura la desviación estándar de la población de la primera muestra.

n ► σ_1 establece el valor de σ_1 para n.

0384 Variable de aplicación σ_2

ID 384 · título recuperado de la página española

Para una prueba o un intervalo que implique la diferencia de dos promedios y una distribución normal, configura la desviación estándar de la población de la segunda muestra.

n ► σ_2 establece el valor de σ_2 para n.

0385 Variable de aplicación s_1

ID 385 · título recuperado de la página española

Establece la desviación estándar de muestra para una prueba de hipótesis o intervalo de confianza. Para una prueba o intervalo que implica la diferencia de dos promedios, establece la desviación estándar de muestra de la primera muestra.

n ► s_1 establece el valor de s_1 para n.

0386 Variable de aplicación s_2

ID 386 · título recuperado de la página española

Para un examen o intervalo que implica la diferencia de dos promedios, establece la desviación estándar de muestra de la segunda muestra.

n ► s_2 establece el valor de s_2 para n.

0387 Var de aplic. x_1

ID 387 · título recuperado de la página española

Establece el número de éxitos de una prueba de hipótesis o intervalo de confianza de una proporción. Para una prueba o intervalo que implica la diferencia de dos proporciones, establece el número de éxitos de la primera muestra.

n ► x_1 establece el valor de x_1 para n.

0388 Var de aplic. x_2

ID 388 · título recuperado de la página española

Para una prueba o intervalo que implica la diferencia de dos proporciones, establece el número de éxitos de la segunda muestra.

n ► x_2 establece el valor de x_2 para n.

0389 Variable de aplicación n_1

ID 389 · título recuperado de la página española

Establece el tamaño de la muestra para una prueba de hipótesis o un intervalo de confianza. Para una prueba o un intervalo que implica la diferencia de dos promedios o proporciones, establece el tamaño de la primera muestra.

n ► n_1 establece el valor de n_1 para n.

0390 Variable de aplicación n_2

ID 390 · título recuperado de la página española

Para una prueba o intervalo que involucre la diferencia de dos proporciones, establece el tamaño de la segunda muestra.

n ► n_2 establece el valor de n_2 para n.

0391 Variable de aplicación μ_0

ID 391 · título recuperado de la página española

Establece el valor supuesto del promedio de la población para una prueba de hipótesis.

n ► μ_0 establece el valor de μ_0 para n.

0392 Variable de aplicación π_0

ID 392 · título recuperado de la página española

Establece la proporción de éxitos asumida en la prueba Z de una proporción.

n ► π_0 establece el valor de π_0 para n.

0393 Variable de aplicación Pooled

ID 393 · título recuperado de la página española

Determina si las muestras deben agruparse o no para pruebas o intervalos mediante la distribución T de Student con dos promedios.

0 ► Pooled para no agrupados (valor predeterminado)

1 ► Pooled para agrupados

0394 Variable de aplicación ObsList

ID 394 · título recuperado de la página española

ObsList es una lista que contiene los recuentos observados para cada categoría de la última prueba de bondad de ajuste (GoF) Chi cuadrada.

0395 Variable de aplicación ProbList

ID 395 · título recuperado de la página española

ProbList es una lista que contiene las probabilidades para cada categoría de la última prueba de bondad de ajuste (GoF) Chi cuadrado.

0396 Variable de la aplicación Explist

ID 396 · título recuperado de la página española

Explist es una lista que contiene los conteos previstos para cada categoría de la última prueba de bondad de ajuste (GoF) Chi cuadrado.

0397 Variable de aplicación ObsMat

ID 397 · título recuperado de la página española

ObsMat es una matriz que contiene los datos del conteo observado desde el último cálculo de la Prueba de Chi cuadrado de 2 vías.

0399 Variable de aplicación Xlist

ID 399 · título recuperado de la página española

Xlist es una lista que contiene los datos de la variable explicativa (X) de la última inferencia para el cálculo de regresión.

0400 Variable de aplicación Ylist

ID 400 · título recuperado de la página española

Ylist es una lista que contiene los datos de la variable de respuesta (Y) de la última inferencia para el cálculo de regresión.

0401 Variables simbólicas

ID 401 · título localizado en HelpEs

Hay cuatro variables simbólicas de aplicación, cada una de las cuales corresponde a uno de los cuatro campos posibles en la vista simbólica de la aplicación Inferencia:

- Método
- Tipo
- AltHyp
- DataType

0402 Variable de aplicación InfType

ID 402 · título recuperado de la página española

InfType determina el tipo de prueba de hipótesis, intervalo de confianza, prueba Chi-cuadrado o inferencia para cálculo de regresión. Su función depende del valor del método de la variable.

Con Método=0 para las pruebas de hipótesis, los valores constantes y sus significados son los siguientes:

- InfType:= 0 para la prueba de Z: 1 promedio
- InfType:= 1 para la prueba de Z: 2 promedios
- InfType:= 2 para la prueba de Z: 1 proporción
- InfType:= 3 para la prueba de Z: 2 proporción
- InfType:= 4 para la prueba de T: 1 promedio
- InfType:= 5 para la prueba de T: 2 promedios

Con Método=1 para intervalos de confianza, los valores constantes y sus significados son los siguientes:

- InfType:= 0 para Int Z: 1 promedio
- InfType:= 1 para Int Z: 2 promedios
- InfType:= 2 para Int Z: 1 proporción
- InfType:= 3 para Int Z: 2 proporciones
- InfType:= 4 para Int T: 1 promedio
- InfType:= 5 para Int T: 2 promedios

Con Método=2 para pruebas de Chi cuadrado, los valores constantes y sus significados son los siguientes:

- InfType:= 0 para GoF de Chi cuadrado
- InfType:= 1 para la prueba de Chi cuadrado de dos vías

Con Método=3 para inferencia de regresión, los valores constantes y sus significados son los siguientes:

- InfType:= 0 para prueba t de regresión lineal
- InfType:= 1 para el intervalo de confianza para la pendiente
- InfType:= 2 para el intervalo de confianza para interceptación
- InfType:= 3 para el intervalo de confianza para respuesta promedio
- InfType:= 4 para el intervalo de predicción para una futura respuesta

0403 Variable de aplicación Method

ID 403 · título recuperado de la página española

Method determina si la aplicación Inferencia se ha establecido para calcular los resultados de la prueba de hipótesis, intervalos de confianza, pruebas de Chi cuadrado o inferencia para los cálculos de regresión.

- Method := 0 para pruebas de hipótesis
- Method := 1 para intervalos de confianza
- Method := 2 para pruebas de Chi cuadrado
- Method := 3 para inferencia de regresión

0404 Variable de aplicación AltHyp

ID 404 · título recuperado de la página española

AltHyp determina la hipótesis alternativa que se usa para pruebas de hipótesis.

- AltHyp := 0 para $\mu < \mu_0$
- AltHyp := 1 para $\mu > \mu_0$
- AltHyp := 2 para $\mu \neq \mu_0$

0405 Variable de aplicación DataType

ID 405 · título recuperado de la página española

Para la prueba de bondad de ajuste (GoF) de Chi cuadrado, DataType determina si la lista esperada contiene probabilidades o conteos.

DataType:= 0 para datos de conteo

DataType:= 1 para probabilidades

0406 Funciones de la aplicación Inferencia

ID 406 · título localizado en HelpEs

En esta sección se enumeran las funciones específicas de la aplicación Inferencia.

0407 Función de la aplicación DoInference

ID 407 · título recuperado de la página española

Sintaxis

DoInference()

Calcula un intervalo de confianza o contrasta una hipótesis.

Realiza los mismos cálculos que al tocar [[Calc]] en la vista Numérica de la aplicación Inferencia y guarda los resultados en las variables de resultados correspondientes de la aplicación Inferencia. Los resultados dependen del contenido de las variables Method, Type y AltHyp de la vista Simbólica de la aplicación Inferencia.

0408 Prueba Z de una muestra para un promedio

ID 408 · título recuperado de la página española

Sintaxis

HypZlmean(PromMuest, TamMuest, PromPobNul, DesEstPob, NivSeg, Modo)

Modo: Especifica qué hipótesis alternativa utilizar:

- 1: Menor que
- 2: Mayor que
- 3: No es igual

Devuelve una lista que contiene (en orden):

- Rechazar (0) o no rechazar (1) la hipótesis nula
- Valor de la prueba Z
- Valor promedio de muestra de entrada
- Probabilidad de cola superior
- Valor Z crítico superior asociado con el nivel α de entrada
- Valor crítico de la estadística asociado al valor Z crítico

0409 Prueba Z de dos muestras para promedios

ID 409 · título recuperado de la página española

Sintaxis

HypZ2mean(PromMuest, PromMuest2, TamMuest, TamMuest2, DesEstPob, DesEstPob2, NivSeg, Modo)

Modo: Especifica qué hipótesis alternativa utilizar:

- 1: Menor que
- 2: Mayor que
- 3: No es igual

Devuelve una lista que contiene (en orden):

- Rechazar (0) o no rechazar (1) la hipótesis nula
- Valor de la prueba Z
- Valor de la prueba $\Delta\bar{x}$
- Probabilidad de cola superior
- Valor Z crítico superior asociado con el nivel α de entrada
- Valor crítico de $\Delta\pi$ asociado al valor Z crítico

0410 Prueba z de una muestra para proporción

ID 410 · título recuperado de la página española

Sintaxis

HypZ1prop(RecCorr, TamMuest, PropPobNu1, NivSeg, Modo)

Modo: Especifica qué hipótesis alternativa utilizar:

- 1: Menor que
- 2: Mayor que
- 3: No es igual

Devuelve una lista que contiene (en orden):

- Rechazar (0) o no rechazar (1) la hipótesis nula
- Valor de la prueba Z
- Valor de la prueba π
- Probabilidad de cola superior
- Valor Z crítico superior asociado con el nivel α de entrada
- Valor crítico de π asociado al valor Z crítico

0411 Prueba Z de dos muestras para proporciones

ID 411 · título recuperado de la página española

Sintaxis

HypZ2prop(RecCorr1, RecCorr2, TamMuest1, TamMuest2, NivSeg, Modo)

Modo: Especifica qué hipótesis alternativa utilizar:

- 1: Menor que
- 2: Mayor que
- 3: No es igual

Devuelve una lista que contiene (en orden):

- Rechazar (0) o no rechazar (1) la hipótesis nula
- Valor de la prueba Z
- Valor de la prueba $\Delta\pi$
- Probabilidad de cola superior
- Valor Z crítico superior asociado con el nivel α de entrada
- Valor crítico de $\Delta\pi$ asociado al valor Z crítico

0412 Prueba T de una muestra para un promedio

ID 412 · título recuperado de la página española

Sintaxis

HypT1mean(PromMuest, DesEstMuest, TamMuest, PropPobNu1, NivSeg, Modo)

Modo: Especifica qué hipótesis alternativa utilizar:

- 1: Menor que
- 2: Mayor que
- 3: No es igual

Devuelve una lista que contiene (en orden):

- Rechazar (0) o no rechazar (1) la hipótesis nula
- Valor de la prueba T
- Entrada de valor ☐
- Probabilidad de cola superior
- Grados de libertad
- Valor T crítico superior asociado con el nivel α de entrada

- Valor crítico de la estadística asociada al valor T crítico

0413 Prueba t de dos muestras para promedios

ID 413 · título recuperado de la página española

Sintaxis

HypT2mean(PromMuest1, PromMuest2, DesEstMuest1, DesEstMuest2, TamMuest1, TamMuest2, NivSeg, Pooled, Modo)

Pooled: especifica si las muestras se agrupan o no:

- 0: no se agrupan
- 1: se agrupan

Modo: Especifica qué hipótesis alternativa utilizar:

- 1: Menor que
- 2: Mayor que
- 3: No es igual

Devuelve una lista que contiene (en orden):

- Rechazar (0) o no rechazar (1) la hipótesis nula
- Valor de la prueba T
- Valor de la prueba Δ
- Probabilidad de cola superior
- Los grados de libertad
- Valor T crítico superior asociado con el nivel α de entrada
- Valor crítico de Δ asociado al valor T crítico

0414 CI normal de una muestra para un promedio

ID 414 · título recuperado de la página española

Sintaxis

ConfZ1mean(PromMuest, TamMuest, DesEsPobm, NivConf)

Intervalo de confianza normal de una muestra para un promedio.

Devuelve una lista que contiene (en orden):

- Menor valor Z crítico
- Límite inferior del intervalo de confianza
- Límite superior del intervalo de confianza

0415 CI normal de dos muestras para un promedio

ID 415 · título recuperado de la página española

Sintaxis

ConfZ2mean(PromMuest1, PromMuest2, TamMuest1, TamMuest2, DesEstPob1, DesEstPob2, NivConf)

Intervalo de confianza normal de dos muestras para la diferencia de dos promedios.

Devuelve una lista que contiene (en orden):

- Menor valor Z crítico
- Límite inferior del intervalo de confianza
- Límite superior del intervalo de confianza

0416 CI normal de una muestra para una proporción

ID 416 · título recuperado de la página española

Sintaxis

ConfZ1prop(RecCorr, TamMuest, NivConf)

Intervalo de confianza normal de una muestra para una proporción.

Devuelve una lista que contiene (en orden):

- Menor valor Z crítico
- Límite inferior del intervalo de confianza
- Límite superior del intervalo de confianza

0417 CI normal de dos muestras para proporciones

ID 417 · título recuperado de la página española

Sintaxis

`ConfZ2prop(RecCorr1, RecCorr2, TamMuest1, TamMuest2, NivConf)`

Intervalo de confianza normal de dos muestras para la diferencia de dos proporciones.

Devuelve una lista que contiene (en orden):

- Menor valor Z crítico
- Límite inferior del intervalo de confianza
- Límite superior del intervalo de confianza

0418 Intervalo t de una muestra para la media

ID 418 · título recuperado de la página española

Sintaxis

`ConfT1mean(PromMuest, DesEstMuest, TamMuest, NivConf)`

Intervalo de confianza para una prueba T de estudiante de una muestra promedio.

Devuelve una lista que contiene (en orden):

- Grados de libertad
- Menor valor t crítico
- Límite inferior del intervalo de confianza
- Límite superior del intervalo de confianza

0419 Intervalo t de dos muestras para las medias

ID 419 · título recuperado de la página española

Sintaxis

`ConfT2mean(PromMuest, PromMuest2, DesEstMuest, DesEstMuest2, TamMuest, TamMuest2, Pooled, NivConf)`

Intervalo de confianza para una prueba T de dos muestras para la diferencia de dos promedios.

Pooled: especifica si las muestras se agrupan o no:

- 0: no se agrupan
- 1: se agrupan

Devuelve una lista que contiene (en orden):

- Los grados de libertad
- El menor valor crítico de t
- Límite inferior del intervalo de confianza
- Límite superior del intervalo de confianza
- El punto medio del intervalo

0420 χ^2 Bondad de ajuste

ID 420 · título recuperado de la página española

Sintaxis

`Chi2GOF(List1, List2, Value [, DegreeFreedom])`

La función Chi2GOF (prueba de bondad de ajuste chi-cuadrado) se utiliza para realizar una prueba de bondad de ajuste chi-cuadrado. Esta prueba compara los datos observados con los datos esperados para determinar si existen diferencias significativas entre ellos.

Parámetros

- List1: Una lista de datos de conteo observados.
- List2: Si Value=0, esta lista se toma como una lista de probabilidades esperadas. Si Value=1, se toma como una lista de conteos esperados.
- Value: Un valor de 0 o 1 que especifica cómo interpretar List2. Con Value=0, List2 se trata como una lista de probabilidades esperadas; con Value=1, como una lista de conteos esperados.
- DegreeFreedom (opcional): Los grados de libertad para la prueba. Si no se especifica, el valor predeterminado es el número de elementos en las listas menos uno.

Devuelve una lista que contiene:

- Valor de la estadística χ^2 : Una medida de cuánto difieren los conteos observados de los esperados.
- Probabilidad: El valor p de la prueba, que indica la probabilidad de obtener tales diferencias por casualidad.
- Grados de libertad: El número de grados de libertad utilizados en la prueba.

0421 χ^2 Prueba de dos vías

ID 421 · título recuperado de la página española

Sintaxis

Chi2TwoWay(Matriz)

Dada una matriz de datos de recuento, devuelve una lista que contiene:

- Valor estadístico de χ^2
- Probabilidad
- Grados de libertad

0422 Prueba t de regresión lineal

ID 422 · título recuperado de la página española

Sintaxis

LinRegrTTest(ListX, ListY, AltHyp)

Realiza una prueba t para la pendiente de la recta de regresión lineal.

Parámetros

- ListX: una lista de valores de la variable explicativa
- ListY: una lista de valores de la variable respuesta
- AltHyp: un entero que especifica la hipótesis alternativa

El argumento AltHyp determina la hipótesis alternativa usada para la prueba:

- AltHyp := 0 para $\beta_1 < 0$
- AltHyp := 1 para $\beta_1 > 0$
- AltHyp := 2 para $\beta_1 \neq 0$

Devuelve una lista que contiene los siguientes valores en este orden:

- t, el valor de la prueba
- P, el valor p asociado con el resultado de la prueba
- DF, los grados de libertad
- β_0 , la intersección de la recta de regresión lineal
- β_1 , la pendiente de la recta de regresión lineal
- s, el error estándar alrededor de la recta
- SE(β_1), el error estándar de la pendiente
- SE(β_0), el error estándar de la intersección
- r, el coeficiente de correlación
- R^2 , el coeficiente de determinación

0423 CI de regresión lineal para Pendiente

ID 423 · título recuperado de la página española

Sintaxis

`LinRegrTConfSlope(ListX, ListY, ConfidenceLevel)`

Calcula un intervalo de confianza para la pendiente de la recta de regresión lineal.

Parámetros

- ListX: una lista de valores de la variable explicativa
- ListY: una lista de valores de la variable de respuesta
- ConfidenceLevel: el nivel de confianza para el intervalo

Devuelve una lista que contiene los siguientes valores en este orden:

- El nivel de confianza dado (ConfidenceLevel)
- t, el valor crítico asociado al nivel de confianza dado
- DF, los grados de libertad
- β_1 , la pendiente de la recta de regresión lineal
- $SE(\beta_1)$, el error estándar de la pendiente
- El límite inferior del intervalo de confianza para la pendiente
- El límite superior del intervalo de confianza para la pendiente

0424 CI de regresión lineal para la interceptación

ID 424 · título recuperado de la página española

Sintaxis

`LinRegrTConfInt(ListX, ListY, ConfidenceLevel)`

Calcula un intervalo de confianza para la ordenada en el origen de la recta de regresión lineal.

Parámetros

- ListX: una lista de valores de la variable explicativa
- ListY: una lista de valores de la variable de respuesta
- ConfidenceLevel: el nivel de confianza para el intervalo

Devuelve una lista que contiene los siguientes valores en este orden:

- El nivel de confianza dado (ConfidenceLevel)
- t, el valor crítico asociado con el nivel de confianza dado
- DF, los grados de libertad
- β_0 , la ordenada en el origen de la recta de regresión lineal
- $SE(\beta_0)$, el error estándar de la ordenada en el origen
- El límite inferior del intervalo de confianza para la ordenada en el origen
- El límite superior del intervalo de confianza para la ordenada en el origen

0425 CI de regresión lineal para respuesta promedio.

ID 425 · título recuperado de la página española

Sintaxis

`LinRegrTMeanResp(ListX, ListY, ValueX, ConfidenceLevel)`

Computes a confidence interval for the mean response at a given X-value.

Parámetros

- ListX: a list of explanatory variable values
- ListY: a list of response variable values
- ValueX: the explanatory variable value at which the mean response is estimated
- ConfidenceLevel: the confidence level for the interval

Returns a list containing the following values in this order:

- The given X-value (ValueX)
- The given confidence level (ConfidenceLevel)
- t, the critical value associated with the given confidence level

- DF, the degrees of freedom
- $\hat{Y}$, the estimated mean response for the given X-value
- $SE(\hat{Y})$, the standard error of the mean response
- The lower bound of the confidence interval for the mean response
- The upper bound of the confidence interval for the mean response

0426 Intervalo de predicción de regresión lineal

ID 426 · título recuperado de la página española

Sintaxis

`LinRegrTPredInt(ListX, ListY, ValueX, ConfidenceLevel)`

Calcula un intervalo de predicción para una respuesta futura en un valor X dado.

Parámetros

- ListX: una lista de valores de la variable explicativa
- ListY: una lista de valores de la variable de respuesta
- ValueX: el valor de la variable explicativa en el que se predice la respuesta futura
- ConfidenceLevel: el nivel de confianza para el intervalo

Devuelve una lista que contiene los siguientes valores en este orden:

- El valor X dado (ValueX)
- El nivel de confianza dado (ConfidenceLevel)
- t, el valor crítico asociado con el nivel de confianza dado
- DF, los grados de libertad
- $\hat{Y}$, la respuesta predicha para el valor X dado
- $SE(\hat{Y})$, el error estándar de la respuesta predicha
- El límite inferior del intervalo de predicción para la respuesta futura
- El límite superior del intervalo de predicción para la respuesta futura

0427 ANOVA de una vía

ID 427 · título recuperado de la página española

Sintaxis

`AnovaOneWay(List1, List2 [, List3, ..., List14])`

Realiza un análisis de varianza de una vía utilizando hasta 14 grupos de tratamiento.

Parámetros

- List1, List2: los grupos de tratamiento
- List3, ..., List14: grupos de tratamiento adicionales opcionales

Devuelve una lista que contiene los siguientes valores en este orden:

- F, el valor F
- P, la probabilidad asociada al valor F
- DF, los grados de libertad de los tratamientos
- SS, la suma de cuadrados de los tratamientos
- MS, el cuadrado medio de los tratamientos
- DFerr, los grados de libertad de los errores
- SSerr, la suma de cuadrados de los errores
- MSerr, el cuadrado medio de los errores

0428 Aplicación Data Streamer

ID 428 · título localizado en HelpEs

La aplicación DataStreamer simplifica la recogida de datos de los sensores a través de HP StreamSmart 410. La aplicación DataStreamer recopila los datos y, a continuación, permite identificar el conjunto de datos exacto que

desea enviar a las aplicaciones 1-Var estadística o 2-Var estadística para su análisis y modelado.

Pulse **[Inicio]** o presione Enter para iniciar la aplicación.

Para obtener más información sobre la aplicación, consulte la guía de usuario de StreamSmart 410.

0429 Vista de gráfico de Data Streamer

ID 429 · título localizado en HelpEs

Presione Plot para volver a esta vista en cualquier momento. La Vista de gráfico es la predeterminada de esta aplicación. En esta vista, se pueden ver hasta cuatro transferencias de datos, una por cada canal activo de HP StreamSmart 410.

Las teclas de menú son:

- **[CANAL]** permite seleccionar un canal (transferencia) para realizar un trazado, etc.
- **[DESPL.]/[ZOOM]** permite alternar entre barridos (desplazamientos) y acercar o alejar la imagen con las teclas de dirección DESPL.: permite desplazarse hacia arriba, hacia abajo, a la izquierda y a la derecha. ZOOM: permite acercar o alejar la imagen en vertical o en horizontal
- **[ÁMBITO]** permite cambiar al modo de osciloscopio
- **[INICIO]/[FINAL]** permite iniciar o detener una transferencia de datos nueva

Con la tecla de menú **[DESPL.]** activada, utilice la rueda hacia arriba/abajo para centrar la transferencia activa en la ventana de gráficos.

Presione la tecla de menú **[DESPL.]/[ZOOM]** para cambiar a ZOOM. Ahora use la rueda hacia la izquierda/derecha para disminuir o acelerar las transferencias, respectivamente.

Tanto el paneo como el zoom se pueden hacer mientras se transmiten los datos para obtener una experiencia interactiva.

Presione la tecla de menú **[FINAL]** y, a continuación, la tecla de menú **[EXPRT]** para aislar los datos deseados y exportarlos a las aplicaciones 1-Var estadística o 2-Var estadística para su análisis.

Presione Num para abrir la Vista numérica de la aplicación.

0430 Vista numérica de Data Streamer

ID 430 · título localizado en HelpEs

Presione Num para volver a esta vista en cualquier momento. En la Vista numérica, se muestran los datos de entrada en forma numérica, en vez de gráfica. Esta vista numérica sirve para la supervisión numérica de los datos, como al realizar determinados experimentos.

Botones de menú:

- **[Agreg.]** permite agregar una lectura al conjunto de datos actual tal como se especifica en **[Conf.]**.
- **[Conf.]** permite seleccionar el tipo de experimento y el destino de los datos
- **[1s]** permite seleccionar la frecuencia de actualización del sensor: 0.5, 1 o 1.5 segundos (el valor predeterminado es 1 segundo)
- **[Estado]** permite acceder directamente a la aplicación de estadística (tal como se especifica en **[Conf.]**) para visualizar y analizar el conjunto de datos actual.

Toque **[Conf.]** para definir el tipo de experimento con eventos seleccionados que desea realizar y para seleccionar el destino de la aplicación y las columnas para el conjunto de datos final. En la vista de configuración hay campos y botones de menú para definir el experimento con eventos seleccionados, tal como se define a continuación.

Campos:

- Apl.: permite seleccionar Var 1 estadística o Var 2 estadística como el destino para los datos (aparecerán otras aplicaciones si ha guardado una versión de las aplicaciones de estadística)
- Método: permite seleccionar solo eventos o eventos con entrada. Si selecciona eventos con entrada, cada vez que pulse el botón de menú **[Agreg.]** en la Vista numérica, se le solicitará que agregue una entrada numérica para el punto de datos.
- Entrada: permite seleccionar una columna en la aplicación de estadística que seleccione para su entrada (solo el método de experimento de eventos con entrada)
- CN1-CN4: seleccione columnas en su aplicación de estadística preferida para su(s) conjunto(s) de datos.

Botones de menú:

- [[Selec.]] permite seleccionar una aplicación, un método, una columna, etc.
- [[✓]] permite seleccionar o anular la selección de un canal para la exportación
- [[Canc.]] permite volver a la Vista numérica sin guardar los cambios
- [[OK]] permite guardar los cambios y volver a la Vista numérica.

0431 Aplicación Solucionador

ID 431 · título localizado en HelpEs

La aplicación Solucionador permite definir hasta diez expresiones con tantas variables como desee en cada una. Puede calcular una única expresión para una de sus variables, en función de un valor de inicialización. También puede calcular un sistema de ecuaciones (lineal o no lineal).

Si dos o más de sus ecuaciones comparten una o varias variables, los valores actuales o calculados de estas variables se trasladarán cuando pase de una ecuación a otra.

Para iniciar la aplicación Solucionador, vaya a la biblioteca de aplicaciones y pulse el icono de la aplicación Solucionador. También puede utilizar la rueda para seleccionar el icono de la aplicación Solucionador, luego pulse [[Inicio]] o presione Enter para iniciar la aplicación.

0432 Vista simbólica de Solucionador

ID 432 · título localizado en HelpEs

Utilice esta vista para introducir y editar hasta diez expresiones, denominadas E0 a E9. Cada expresión puede utilizar cualquier variable definida, incluidas A a Z y θ . Resalte uno de los diez campos y comience a introducir una expresión, o toque [[Editar]] para modificar una expresión existente.

Cada expresión tiene un selector de color a la izquierda de su campo. Toque el cuadro de color para elegir el color del gráfico de esa expresión.

Los botones del menú son:

- [[Editar]] abre un cuadro de entrada para editar la expresión seleccionada
- [[✓]] selecciona una expresión para resolver
- [[=]] inserta un signo igual
- [[Mostr.]] muestra la expresión resaltada en modo de pantalla completa, con desplazamiento horizontal y vertical activado
- [[Eval.]] resuelve referencias cuando una expresión está definida en función de otra

Presione Num para mostrar la vista numérica, donde se realiza la resolución.

0433 Vista de gráfico de Solucionador

ID 433 · título localizado en HelpEs

La Vista de gráfico ofrece una representación gráfica de la expresión seleccionada en la Vista simbólica, si solo se ha seleccionado una expresión. Los lados izquierdo y derecho de la expresión actual se trazan como dos gráficos independientes. La variable resaltada en la Vista numérica se entiende como una variable independiente con fines de creación de gráficos. El punto o los puntos donde estas dos gráficas se cruzan son soluciones a la ecuación. Si no hay "lado derecho", "=0" se utiliza como el lado derecho implícito.

Los botones del menú son:

- [[Menú]] muestra y oculta el menú Gráfico, con opciones de zoom y trazado.
- [[Zoom]] permite abrir el menú Zoom, con opciones para acercar o alejar el zoom
- [[Trazar]] permite activar y desactivar el cursor de trazado
- [[Ir a]] coloca el cursor de trazado en el punto de la función con un determinado valor x
- [[Defn]] muestra la definición simbólica de la función actual

Si se selecciona más de una ecuación en la Vista simbólica, la Vista de gráfico no está disponible.

0434 Configuración de gráfico de Soluc.

ID 434 · título localizado en HelpEs

La pantalla Config. de gráfico Solucionador permite controlar la apariencia de la ventana del gráfico, incluida la apariencia del cursor, si deben dibujarse o no los ejes, etc. Esta configuración tiene tres páginas.

En la primera página de configuración, los campos son los siguientes:

- Rng X: el rango horizontal del gráfico
- Rng Y: el rango vertical del gráfico
- Mrc X: espaciado horizontal entre marcas de graduación
- Mrc Y: espaciado vertical entre marcas de graduación

Los botones del menú en la primera página son:

- [[Editar]] permite editar el valor que aparece en el campo seleccionado
- [-Página 1/3 ▼-] permite ir a la segunda página de la configuración

En la segunda página de configuración, los campos son los siguientes:

- Ejes: activa o desactiva los ejes
- Etiquetas: activa o desactiva las etiquetas de los ejes
- Puntos cuadr.: activa o desactiva los puntos de la cuadrícula
- Líneas cuadríc.: activa o desactiva las líneas de la cuadrícula
- Cursor: permite elegir entre Estándar, Inversión y Parpadeando
- Método: permite elegir entre Adaptable, Segm. de increm. fijo y Ptos de increm. fijo

Los botones del menú en la segunda página son:

- [[✓]] permite activar o desactivar la configuración actual
- [[Selec.]] permite elegir una opción de un cuadro de selección
- [-▲ Página 2/3 ▼-] permite volver a la primera página de la configuración o ir a la tercera página

El campo Método requiere una explicación. De forma predeterminada, la HP Prime utiliza el método Adaptable, un método avanzado que proporciona resultados muy precisos. Puede elegir el método más tradicional, denominado Segm. de increm. fijo, que muestrea valores de x, calcula sus valores y correspondientes y, a continuación, traza los puntos y los conecta. O bien puede seleccionar Ptos de increm. fijo, que funciona como Segm. de increm. fijo, pero no conecta los puntos.

En la tercera página de configuración, puede configurar una imagen de fondo opcional para la vista de gráfico. Los elementos son los siguientes:

- El primer campo, sin etiqueta: permite elegir cómo se muestra la imagen de fondo. Las opciones disponibles son Sin fondo, Centrado, Alargada, Mejor ajuste y Rango XY.
- Imagen: permite elegir la imagen que se utilizará como fondo.
- Opacidad: permite establecer el nivel de transparencia de la imagen de fondo seleccionada. 0 indica transparencia total.

Los botones del menú en la tercera página son:

- [[Selec.]] permite elegir una opción de un cuadro de selección. Según el campo seleccionado, este botón permite seleccionar el modo de visualización del fondo o elegir una de las imágenes disponibles.
- [-▲ Página 3/3-] permite volver a la página anterior de la configuración
- [[Impor.]] permite importar archivos de imagen desde cualquier aplicación que contenga imágenes compatibles

El botón [[Impor.]] abre el cuadro de diálogo Mi imagen. En este cuadro de diálogo, el cuadro de selección Apl. permite seleccionar cualquier aplicación que contenga archivos de imagen compatibles. Después de seleccionar una aplicación y tocar [[OK]], la configuración vuelve a la tercera página. Las imágenes de la aplicación seleccionada quedan disponibles en el campo Imagen.

0435 Vista numérica de Solucionador

ID 435 · título localizado en HelpEs

La Vista numérica de Solucionador sirve para introducir los valores de todas las variables conocidas y calcular las desconocidas.

Si solo se activa una ecuación en la Vista numérica, introduzca los valores de las variables conocidas y, a continuación, seleccione la variable desconocida y toque [[Solucionador]]

Si se activa más de una ecuación en la Vista numérica, introduzca los valores de las variables conocidas, active las variables que desee calcular y toque [[Solucionador]]

Tenga en cuenta que los valores actuales para la variable desconocida se pueden usar como inicialización para el algoritmo de cálculo.

Los botones del menú son:

- **[[Editar]]** permite editar el valor de la variable actual
- **[[Info]]** permite obtener información sobre una solución
- **[[Defn]]** muestra la expresión que se está calculando
- **[[Soluc.]]** calcula la variable seleccionada actualmente

Nota: Si Info contiene el mensaje Extremum, esto indica que es muy probable que no haya solución para la ecuación o el sistema.

0436 Variables de la aplicación Soluc.

ID 436 · título localizado en HelpEs

Para mostrar las variables relacionadas con la aplicación Solucionador, pulse Vars, toque **[[Apl.]]** y seleccione Soluc.

La aplicación Solucionador incluye variables en las categorías siguientes:

- Gráfico (consulte Variables de aplicaciones comunes)
- Modos (consulte Variables de aplicaciones comunes)

Variables de la Vista simbólica

Las variables de la Vista simbólica de la aplicación Solucionador son E1 a E9 y E0. Pueden contener cualquier expresión. Para seleccionar la variable independiente, se resalta en la Vista numérica.

En := expresión, donde n es un número entero entre 0 y 9, inclusive

Ejemplos

E3:='A+2*X=3*B'

0437 Función de aplicación Solucionador

ID 437 · título recuperado de la página española

Sintaxis

SOLVE(En, Var [, Guess])

Resuelve una ecuación o expresión para la variable Var.

SOLVE solo funciona cuando la aplicación Solucionador está activa.

El argumento En puede ser una ecuación, una expresión o el nombre de una de las variables simbólicas de Solucionador E0-E9. Si En es una ecuación, SOLVE busca un valor de Var que satisfaga la ecuación. Si En es una expresión, SOLVE busca un valor de Var que haga que la expresión sea igual a cero.

El valor opcional Guess se usa como valor inicial para el algoritmo numérico de resolución.

SOLVE también puede devolver una lista que contiene un mensaje de información y uno o más números cuando no se encuentra una solución ordinaria.

Ejemplos

[Ejemplo gráfico E1]

[Ejemplo gráfico E2] devuelve {Error: Extremum found, -1.57079487496} porque $\sin(x)+2$ no puede ser igual a cero. El número devuelto es la ubicación de un extremo, no una solución.

0438 Aplicación Solucionador lineal

ID 438 · título localizado en HelpEs

La aplicación Solucionador lineal soluciona sistemas lineales de 2 ecuaciones en 2 variables o de 3 ecuaciones en 3 variables.

Para iniciar la aplicación Solucionador lineal, vaya a la biblioteca de aplicaciones y pulse el icono de la aplicación Solucionador lineal. También puede utilizar la rueda para seleccionar el icono de la aplicación Solucionador lineal, luego pulse **[[Inicio]]** o presione Enter para iniciar la aplicación.

0439 Vista numérica de Solucionador lineal

ID 439 · título localizado en HelpEs

La aplicación Solucionador lineal se abre en la Vista numérica, la única vista de esta aplicación. De forma predeterminada, la aplicación inicia sistemas de ecuaciones lineales 3×3 listos para solucionar. Tenga en cuenta el punto en el botón del menú $[[3 \times 3]]$ para indicar que se encuentra activo. Toque el botón de menú $[[2 \times 2]]$ para calcular sistemas de ecuaciones lineales 2×2.

Introduzca los coeficientes de cada variable en cada ecuación, así como el término constante. La solución al sistema aparece en tiempo real en la parte inferior de la pantalla.

Los botones del menú son:

- $[[\text{Editar}]]$ abre una línea para editar el valor elegido
- $[[2 \times 2]]$ permite solucionar un sistema 2×2 de 2 ecuaciones lineales con 2 variables
- $[[3 \times 3]]$ permite solucionar un sistema 3×3 de 3 ecuaciones lineales con 3 variables

0440 Variables de Solucionador lineal

ID 440 · título localizado en HelpEs

Además de las variables de modo (que son comunes para todas las aplicaciones), la aplicación Solucionador lineal incluye dos variables:

- LSystem
- LSolution

0441 Variable de aplicación LSystem

ID 441 · título recuperado de la página española

Contiene una matriz 2×3 o 3×4 que representa un sistema lineal 2×2 o 3×3.

matriz ► LSystem, donde matriz hace referencia a una matriz o al nombre de la matriz de variables de M0 a M9.

0442 Variable de aplicación LSolution

ID 442 · título recuperado de la página española

Contiene un vector con la última solución encontrada por la aplicación Solucionador lineal o la función de aplicación LSolve.

0443 Funciones de la aplicación Solucionador lineal

ID 443 · título localizado en HelpEs

Esta sección enumera las funciones específicas de la aplicación Solucionador lineal.

0444 Función de aplicación Solve2×2

ID 444 · título recuperado de la página española

Sintaxis

Solve2x2(a, b, C, d, e, f)

Resuelve el sistema lineal de 2 x 2 representado por:

$$ax+by=c$$

$$dx+ey=f$$

0445 Función de aplicación Solve3×3

ID 445 · título recuperado de la página española

Sintaxis

Solve3x3(a, b, C, d, e, f, g, h, i, j, k, l)

Resuelve el sistema lineal 3 x 3 representado por:

$$ax+by+cz=d$$

$$ex+fy+gz=h$$

$$ix+jy+kz=l$$

0446 Función de aplicación LinSolve

ID 446 · título recuperado de la página española

Sintaxis

LinSolve(matriz)

Solucionador sistema lineal. Resuelve el sistema lineal $N \times N$ representado por una matriz $N \times (N+1)$. LinSolve([[a,b,c],[d,e,f]]) resuelve el sistema lineal:

$$ax+by=c$$

$$dx+ey=f$$

0457 Aplicación Solucionador de triángulo

ID 457 · título localizado en HelpEs

Un triángulo cuenta con 3 lados, cada uno de ellos con una longitud específica, así como con 3 ángulos, cada uno de ellos con una medida específica. Al especificar 3 de estos 6 valores se define por completo el triángulo, siempre y cuando uno de ellos sea la longitud de uno de los lados. En el caso ambiguo, la especificación de tres de los valores (incluida la longitud de uno de los lados) define al triángulo en términos de dos alternativas. La aplicación Solucionador de triángulo le permite introducir los 3 valores conocidos (uno de los cuales siempre debe ser la longitud de uno de los lados) y calcular los otros 3, o las dos alternativas de los otros 3.

Para iniciar la aplicación Solucionador de triángulo, vaya a la Biblioteca de aplicaciones y pulse el icono de la aplicación Solucionador de triángulo. También puede utilizar la rueda para seleccionar el icono de la aplicación Solucionador de triángulo, luego pulse [[Inicio]] o presione Enter para iniciar la aplicación. La aplicación Solucionador de triángulo se abre en la Vista numérica, que es la única que tiene esta aplicación.

0458 Vista numérica de Soluc. de triáng.

ID 458 · título localizado en HelpEs

Asegúrese de que el modo de ángulo esté configurado correctamente. Si sus ángulos están en grados, pulse la tecla de menú [[Radianes]] hasta que quede seleccionado Grados.

Los campos son los siguientes:

- a: indica la longitud de un lado del triángulo
- b: indica la longitud de otro lado
- c: indica la longitud del tercer lado
- A: indica la medida del ángulo opuesto al lado a
- B: indica la medida del ángulo opuesto al lado b
- C: indica la medida del ángulo opuesto al lado c

Los botones de menú de la Vista numérica principal son:

- [[Editar]] permite abrir una línea de edición para editar el valor actual de un campo
- [[Grados]]/[[Radianes]] permite cambiar entre los modos de medición de ángulos en grados y radianes
- [[Δ]] permite cambiar entre un solucionador sencillo para triángulos rectángulos y el solucionador general
- [[Soluc.]] permite utilizar los valores actuales para calcular los valores desconocidos

Mediante la rueda, pase a un campo cuyo valor conozca, introduzca el valor y presione Enter. Repita el paso en cada uno de los valores conocidos. Tome en cuenta que uno de los valores debe tener la longitud de un lado. Pulse [[Solucionar]] y la aplicación mostrará las longitudes restantes y las mediciones del ángulo. Si el botón del menú [[Alt]] aparece, significa que hay dos soluciones posibles. Pulse [[Alt]] para alternar entre las dos soluciones.

Si está determinando las propiedades de un triángulo rectángulo, hay disponible un formulario de entrada más sencillo al pulsar el botón de menú con un símbolo de triángulo.

0459 Variables de Soluc. de triángulo

ID 459 · título localizado en HelpEs

Además de las variables de modos (que son comunes en todas las aplicaciones), las variables de la aplicación Solucionador de triángulo corresponden a los campos de la Vista numérica de la aplicación.

0460 Variable de aplicación SideA

ID 460 · título recuperado de la página española

SideA : la longitud del lado opuesto al ángulo A.

n ► SideA, donde $n > 0$, establece el valor de SideA en n.

0461 Variable de aplicación SideB

ID 461 · título recuperado de la página española

SideB : la longitud del lado opuesto al ángulo B.

n ► SideB, donde $n > 0$, establece el valor de SideB en n.

0462 Variable de aplicación SideC

ID 462 · título recuperado de la página española

SideC : la longitud del lado opuesto del ángulo C.

n ► SideC, donde $n > 0$, establece el valor de SideC en n.

0463 Variable de aplicación AngleA

ID 463 · título recuperado de la página española

AngleA : La medida del ángulo A.

El valor de esta variable se interpretará de acuerdo con la configuración de modo de ángulo (Grados o Radianes).

n ► AngleA, donde $n > 0$, establece el valor de AngleA en n.

0464 Variable de aplicación AngleB

ID 464 · título recuperado de la página española

Angle B : La medida del ángulo B.

El valor de esta variable se interpreta según la configuración del modo ángulo (Degrees o Radians) (Grados o Radians).

n ► AngleB, donde $n > 0$, establece el valor de AngleB en n.

0465 Variable de aplicación AngleC

ID 465 · título recuperado de la página española

AngleC : La medida del ángulo C.

El valor de esta variable se interpretará de acuerdo con la configuración de modo de ángulo (Grados o Radianes).

n ► AngleC, donde $n > 0$, establece el valor de AngleC en n.

0466 Variable Tipo de triángulo

ID 466 · título recuperado de la página española

Corresponde al estado de la tecla de menú $[\triangle]$ en la vista Numérica de la aplicación Triangle Solver. Determina si se usa un solucionador de triángulos general o uno de triángulos rectángulos.

0 ► TriType para el solucionador de triángulos general (predeterminado)

1 ► TriType para el solucionador de triángulos rectángulos

0467 Funciones de la aplicación Soluc. de triáng.

ID 467 · título localizado en HelpEs

La aplicación Solucionador de triángulo cuenta con un grupo de funciones que permiten calcular un triángulo completo a partir de la introducción de 3 medidas de este. Los nombres de estos comandos utilizan A para referirse a un ángulo y S para referirse a la longitud de un lado. Para utilizar estos comandos, introduzca tres valores en el

orden especificado por el nombre del comando. Estos comandos devuelven una lista de 6 elementos, formada por los tres argumentos introducidos y los tres valores desconocidos (longitudes de lados y medidas de ángulos).

0468 Función de aplicación AAS

ID 468 · título recuperado de la página española

Sintaxis

AAS(angle, angle, lado)

Toma como argumentos las medidas de dos ángulos y la longitud del lado opuesto al primer ángulo y devuelve una lista que contiene la longitud del lado opuesto al segundo ángulo, la longitud del tercer lado y la medida del tercer ángulo (en ese orden).

Ejemplos

[Ejemplo gráfico E1] (en modo grados)

0469 Función de aplicación ASA

ID 469 · título recuperado de la página española

Sintaxis

ASA(angle, lado, angle)

Toma como argumentos la medida de dos ángulos y la longitud del lado incluido y devuelve una lista que contiene la longitud del lado opuesto del primer ángulo, la longitud del lado opuesto al segundo ángulo y la medida del tercer ángulo (en ese orden)

Ejemplos

[Ejemplo gráfico E1] (en modo grados)

0470 Función de aplicación SAS

ID 470 · título recuperado de la página española

Sintaxis

SAS(lado, angle, lado)

Toma como argumentos la longitud de dos lados y la medida del ángulo incluido y devuelve una lista que contiene la longitud del tercer lado, la medida del ángulo opuesto al tercer lado y la medida del ángulo opuesto al segundo lado.

Ejemplos

[Ejemplo gráfico E1] (en modo grados)

0471 Función de aplicación SSA

ID 471 · título recuperado de la página española

Sintaxis

SSA(lado, lado, angle)

Toma como argumentos las longitudes de dos lados y la medida de un ángulo no incluido y devuelve una lista que contiene la longitud del tercer lado, la medida del ángulo opuesto al segundo lado y la medida del ángulo opuesto al tercer lado..

Nota: En un caso ambiguo, este comando solo le dará una de las dos posibles soluciones.

Ejemplos

[Ejemplo gráfico E1] (en modo grados)

0472 Función de aplicación SSS

ID 472 · título recuperado de la página española

Sintaxis

SSS(lado, lado, lado)

Utiliza como argumentos la longitud de los tres lados de un triángulo para calcular la medida de los tres ángulos opuestos, en orden.

Ejemplos

[Ejemplo gráfico E1] (en modo grados)

0473 Función de la aplicación DoSolve

ID 473 · título recuperado de la página española

Sintaxis

DoSolve()

Resuelve el problema actual en la aplicación Solucionador de triángulo.

Para resolver el problema, la aplicación Solucionador de triángulo debe tener suficientes datos; es decir, debe haber al menos tres valores y uno de ellos debe ser una longitud de lado. Devuelve una lista que contiene los valores desconocidos en la Vista numérica, en orden de aparición en esa vista (de izquierda a derecha y de arriba a abajo).

0474 Aplicación Finanzas

ID 474 · título localizado en HelpEs

La aplicación Finanzas soluciona un conjunto de problemas financieros comunes. En la Vista simbólica, puede seleccionar en una lista de problemas financieros comunes. Luego, puede introducir la Vista numérica para solucionar su problema seleccionado. Algunos de los problemas también tienen una vista de gráfico.

Para iniciar la aplicación Finanzas, vaya a la biblioteca de aplicaciones y pulse el icono de la aplicación Finanzas. También puede utilizar la rueda para seleccionar el icono de la aplicación Finanzas, luego pulse [[Inicio]] o presione Enter para iniciar la aplicación.

1540 Vista simbólica de Finanzas

ID 1540 · título localizado en HelpEs

La Vista numérica de Finanzas le permite elegir cuál cálculo financiero le gustaría realizar en la aplicación Finanzas. Entre ellos se incluyen:

- TVM (valor del dinero en el tiempo): usado para cálculos de interés compuesto que involucran flujos de efectivo regulares y uniformes
- Conversión de interés: hace la conversión entre tasas de interés nominal y efectiva
- Cálculo de fecha: calcula la diferencia entre dos fechas
- Flujo de efectivo: calcula el retorno a la inversión y el valor de los flujos de efectivo
- Depreciación: calcula la disminución del valor del activo en el tiempo
- Umbral de rentabilidad: se usa para encontrar el umbral de rentabilidad entre la cantidad de unidades vendidas, los costos fijos, los costos de fabricación, el precio de venta y el lucro deseado
- Cambio porcentual: calcula un nuevo precio, costo o valor basado en el margen, el incremento de precio, el porcentaje total o el cambio de porcentaje
- Bonos: calcula el rendimiento del bono o el precio del bono
- Black-Scholes: utiliza el modelo matemático Black-Scholes para valorar las opciones europeas de rescate y venta

Primero, seleccione el cálculo que desea en el menú desplegable. Luego, presione Num para introducir la Vista numérica de Finanzas y solucione la ecuación elegida.

Para obtener una descripción más completa de cómo usar cada una de las opciones de finanzas, consulte el texto de ayuda del grupo de la Vista numérica de Finanzas.

0475 Gráfico de amortización de Finanzas

ID 475 · título localizado en HelpEs

El Gráfico de amortización de finanzas muestra gráficamente el plan de amortización. Utilice la rueda a la izquierda/derecha para pasar de un grupo de pagos a otro. Para cada grupo de pagos, el capital y los intereses pagados durante el intervalo se muestran de forma numérica en la parte inferior de la pantalla.

1598 Configuración del gráfico Finanzas

ID 1598 · título localizado en HelpEs

Presione Shift Plot para introducir la Configuración del gráfico Finanzas. La página 1 de la Configuración de gráfico contiene valores que controlan la aparición de gráficos financieros.

En la primera página, los campos son:

- Rng X: el rango horizontal de la ventana de gráfico
- Rng Y: el rango vertical de la ventana de gráfico
- Mrc X: espaciado entre marcas de graduación horizontal
- Mrc Y: espaciado entre marcas de graduación vertical

Los botones del menú en la primera página son:

- [[Editar]] abre un nuevo cuadro de edición para editar el valor que aparece en el campo seleccionado.
- [-Página 1/2 ▼-] muestra la segunda página de la configuración

En la segunda página, los campos son:

- Ejes: activa o desactiva los ejes
- Etiquetas: activa o desactiva las etiquetas del eje
- Puntos cuadr.: activa o desactiva los puntos de la cuadrícula
- Líneas cuadríc.: activa o desactiva las líneas de la cuadrícula
- Cursor: permite elegir entre cursores estándar, invertido y parpadeante

Los botones del menú en la segunda página son:

- [[✓]] activa o desactiva la configuración actual
- [[Selec.]] permite elegir de un cuadro de elección
- [-▲ Página 2/2-] retorna a la primera página de la configuración

0476 Vista numérica de Finanzas

ID 476 · título localizado en HelpEs

La Vista numérica de Finanzas es donde puede encontrar soluciones para problemas financieros de la aplicación Finanzas.

1631 Vista TVM

ID 1631 · título localizado en HelpEs

El Valor del dinero en el tiempo (TVM) le permite resolver problemas de TVM y amortización. Puede realizar cálculos de interés compuesto y crear tablas de amortización.

El interés compuesto es el interés acumulativo, es decir, el interés sobre el interés ya ganado. Los intereses devengados de un capital dado se añaden al capital principal en períodos de capitalización especificados, y luego el monto combinado gana intereses a una determinada tasa. Los cálculos financieros que involucran intereses compuestos incluyen cuentas de ahorro, hipotecas, fondos de pensiones, arrendamientos y anualidades.

Los cálculos TVM se basan en la idea de que un dólar actual valdrá más de un dólar en el futuro. Un dólar actual puede invertirse a un determinado tipo de interés y generar una ganancia que no podría generar el mismo dólar en el futuro. Este principio de TVM describe la idea de las tasas de interés, los intereses compuestos y las tasas de rendimiento.

- N: El número total de pagos o periodos capitalizables por año.
- 1%/Año: La tasa de interés anual nominal (o tasa de inversión). Esta tasa se divide por el número de pagos por año (P/Año) para calcular la tasa de interés nominal por periodo utilizado para el cálculo de interés compuesto. Esta es la tasa de interés utilizada realmente en los cálculos de TVM.
- VA: El valor actual del flujo de efectivo inicial. Para un prestamista o prestatario, el VA es el importe del préstamo; para un inversor, el VA es la inversión inicial. El VA siempre se produce al principio del primer periodo.
- P/Año: El número de pagos realizados en un año.

- IPP: El importe de pago periódico. Los pagos tienen el mismo importe cada periodo y el cálculo de TVM asume que no se omite ningún pago. Los pagos se producen al principio o al final de cada periodo utilizado para el cálculo de interés compuesto, una opción que puede controlar desmarcando o marcando la opción Final.
- C/Año: El número de periodos capitalizables en un año.
- VF: El valor futuro de la transacción: el importe del flujo de efectivo final o el valor liquidado de la serie de flujos de efectivo anteriores. Para un préstamo, se trata del tamaño del pago final global (más allá de cualquier pago habitual debido). Para una inversión, este es el valor al final del periodo de inversión.

Introduzca los valores que conoce, seleccione el importe que desea resolver y toque **[[Soluc]]**. Toque **[[Amort]]** para visualizar y examinar la tabla de amortización para su flujo de caja. Presione Plot para ver el gráfico de amortización.

0477 Tabla de amortización de finanzas

ID 477 · título localizado en HelpEs

La tabla de amortización de finanzas muestra el cronograma de amortización.

El calendario es una tabla que muestra, para cada grupo de pagos, el capital y los intereses pagados durante el grupo, así como el saldo restante al final del grupo.

- Tamaño: perite elegir el tamaño de fuente pequeño, mediano y grande
- TVM: regresa a la vista TVM

Utilice la rueda o arrastre para desplazarse por la tabla.

Pulse la tecla menú **[[TVM]]** para volver a la página TVM cuando termine.

1625 Vista de conversión de interés

ID 1625 · título localizado en HelpEs

La Conversión de interés le permite hacer conversiones entre la tasa de interés nominal (se debe especificar una tasa compuesta después de un periodo dado) y la tasa de interés efectivo (la cantidad de intereses que efectivamente se cobraron a lo largo de un año).

- Nom I%: Tasa de interés nominal: la tasa de interés anual establecida calculada según se representa por P/Año, como por ejemplo 18 % acumulado mensualmente ($P/Año=12$).
- Eff I%: Tasa de interés anual efectiva tomando en cuenta lo acumulado.
- P/Año: Número de periodos anuales en que se acumula la tasa de interés nominal.

1627 Vista cálculo de fecha

ID 1627 · título localizado en HelpEs

Los cálculos de fecha le permiten calcular la diferencia entre dos fechas o calcular una fecha de algún número de días a partir de otra fecha.

- Fecha 1: la primera fecha en formato AAAA.MMDD. Esta fecha debe ser gregoriana y no debe exceder 9999.1231.
- Fecha 2: la segunda fecha en formato AAAA.MMDD. Esta fecha debe ser gregoriana y no debe exceder 9999.1231.
- Diferencia: la diferencia entre las dos fechas en número de días (limitado a más o menos 1.000.000 días ~ 2700 años).
- Cal. 360: especifica que se debe utilizar un calendario de 30 días por mes, 360 días por año para los cálculos. El calendario de 360 días es útil para medir duraciones en los mercados financieros.

Introduzca los datos en cualquiera de los dos campos, resalte el campo restante y pulse **[[Soluc]]**.

1632 Vista de flujo de efectivo

ID 1632 · título localizado en HelpEs

El flujo de efectivo se usa para solucionar los problemas donde los flujos de efectivo ocurren en intervalos regulares. Los problemas con flujos de efectivo regulares, iguales, periódicos se manejan con más facilidad mediante la función TVM.

En la pantalla numérica verá una tabla donde puede introducir los datos del Flujo de efectivo. La parte superior de la tabla tiene campos para introducir estos tres elementos:

- % I de Inver.: inversión o tasa de interés de descuento. La tasa de flujo de efectivo que no necesita ser líquida ni estar altamente disponible, por lo que esta tasa refleja una mayor rentabilidad proporcional al aumento del riesgo.
- % I Seguro: tasa de interés de inversión segura. Esta tasa supone que los fondos necesarios para cubrir los flujos negativos de efectivo se colocan en inversiones que son altamente líquidas y fáciles de retirar a voluntad. Esto hace que estén disponibles "con seguridad" con un riesgo mínimo y por lo tanto un menor rendimiento.
- #FE/Año: el número de flujos de efectivo por año

En la parte inferior de la tabla ingresará sus datos de flujo de efectivo.

- # FE: número que representa la posición del flujo de efectivo en la lista, donde 0 es la inversión inicial. Este número se crea automáticamente al ingresar datos.
- N.º FE: El número de sucesos consecutivos del flujo de efectivo.
- Flujo de efectivo: el importe del flujo de efectivo.

Una vez que haya completado su lista, presione [[Calc]] para ver el análisis de sus datos de flujo de efectivo.

- Tasa Interna de rendimiento (TIR): la tasa de descuento que devuelve un Valor Actual Neto de 0 para los flujos de efectivo ingresados, descontando todos los flujos de efectivo con % I de Inver.
- TIR modificada (MTIR): tasa interna de rendimiento modificada. Un cálculo TIR mejorado que descuenta los flujos de efectivo negativos con % I seguro y flujos de efectivo positivos con % I de Inver.
- TRG Financiero: (TRGF) tasa de rendimiento de la gestión financiera. Un cálculo de TIR más complicado que MIRR, donde los flujos de efectivo negativos son eliminados por los flujos de efectivo positivos anteriores, antes de descontar con % I seguro y luego los flujos de efectivo positivos posteriores se descuentan con % I de Inver.
- Total: la suma de todos los flujos de efectivo, equivalente a NPV si % I de Inver es 0.
- Valor actual neto (VAN): Valor de los flujos de efectivo en el momento del flujo de efectivo inicial, descontando los flujos de efectivo futuros por % I de Inver.
- Valor futuro neto (VFN): valor de los flujos de efectivo en el momento del último flujo de efectivo, descontando los flujos de efectivo anteriores por % I de Inver.
- Serie Neta Uniforme (SNU): pago por período de un flujo de efectivo periódico regular de valor actual equivalente a la lista de flujo de efectivo.
- Reembolso descontado: el número de períodos necesarios para que la inversión devuelva el valor si los flujos de efectivo se descuentan por % I de Inver.
- Reembolso: el número de períodos necesarios para que la inversión devuelva el valor.

1626 Vista de depreciación

ID 1626 · título localizado en HelpEs

La depreciación le permite calcular la pérdida en el valor de los activos a lo largo del tiempo.

El campo Tipo de la Vista simbólica de Finanzas le permite elegir entre los siguientes métodos para calcular la depreciación:

- Línea recta: calcula la depreciación suponiendo que un activo pierde un determinado porcentaje de su valor de forma anual en una cantidad uniformemente distribuida a lo largo de su vida útil.
- Dígitos de suma de los años: un método de depreciación acelerada donde la depreciación en el año y es $(\text{Life-}y + 1)/\text{SOY}$ del activo, donde SOY es la suma de los años de vida del activo. Para un activo con una vida útil de 5 años, $\text{SOY} = 5 + 4 + 3 + 2 + 1 = 15$.
- Saldo decreciente: un método de depreciación acelerada que presume que un activo perderá la mayor parte de su valor durante los primeros años de su vida útil.
- SD con cruce de LR: el saldo decreciente con cruce de línea recta es un método de depreciación acelerada que presupone que un activo perderá la mayor parte de su valor en los primeros años de su vida útil y luego volverá a una depreciación consistente durante la última parte de su vida, calculado con el método de línea recta.
- Línea recta sistema francés: similar al método de línea recta, utilizando la fecha real del calendario en que el activo fue puesto en servicio por primera vez.
- Amortización sistema francés: un método de amortización acelerado con cruce de tipo de línea recta del sistema francés.

Una vez que haya seleccionado el tipo de amortización, ingresará información en los campos siguientes en la Vista numérica de Finanzas:

- Costo: el costo inicial del activo a depreciar.
- Rescate: el valor de rescate del activo al final de su vida útil.

- Vida útil: la vida útil esperada del activo en años.
- Primer uso: el mes (o la fecha para los tipos de depreciación de sistema francés) en que el activo se pone en servicio por primera vez. Nota: el mes se puede introducir con un decimal para indicar el primer uso después del primero del mes. Por ejemplo, si el activo se puso en servicio a mediados de marzo, escriba 3.5.
- Factor: el factor del saldo decreciente como porcentaje. Se utiliza solo para saldo decreciente y saldo decreciente con cruce a línea recta.

Después de ingresar los datos en todos los campos, presione **[[Calc]]** para ver una tabla de resultados calculada comenzando con el primer año y terminando con el último año de vida útil.

- Depreciación: importe de depreciación del ejercicio.
- Valor Depr: valor residual depreciable al final del año.
- Valor de libro: valor residual restante al final del año.

1624 Vista de umbral de rentabilidad

ID 1624 · título localizado en HelpEs

La función de umbral de rentabilidad le permite estudiar problemas que implican una ganancia cuando una cantidad de elementos con un costo de fabricación y un precio fijo para desarrollar y comercializar se vende a un precio determinado. Esta herramienta resuelve la ecuación $\text{Precio de venta} * \text{Cantidad} - \text{Costo} * \text{Cantidad} - \text{Fijo} = \text{Ganancia}$.

- Fijo: costo fijo para desarrollar y comercializar un producto.
- Cantidad: cantidad de unidades vendidas.
- Costo: costo de fabricación o producción por unidad vendida.
- Precio de venta: precio por unidad vendida.
- Ganancia: ganancia esperada.

Introduzca la información conocida en cualquiera de los cuatro campos, mueva el cursor al valor que desee calcular y pulse **[[Soluc]]**.

1628 Vista de cambio de porcentaje

ID 1628 · título localizado en HelpEs

El cambio de porcentaje proporciona dos tipos de herramientas de cálculo de porcentaje: Incremento del precio / Margen o Porcentaje Total / Cambio de Porcentaje.

El campo Tipo en la Vista simbólica de Finanzas le permite elegir entre los siguientes métodos para calcular porcentajes de negocio:

- Incremento del precio / Margen: calcula el incremento del precio como porcentaje del costo o el margen como un porcentaje del precio.
- Total / Cambio: calcula el nuevo valor basado en el porcentaje total del valor anterior o basado en el cambio porcentual del valor anterior

Incremento del precio / Margen usa las siguientes entradas:

- Costo: costo total de comprar o fabricar el elemento
- Precio: precio de ventas del elemento
- Incremento de precio: un porcentaje del costo: $((\text{Precio} - \text{Costo}) / \text{Costo}) * 100$
- Margen: un porcentaje del precio: $((\text{Precio} - \text{Costo}) / \text{Precio}) * 100$

Total / Cambio usa las siguientes entradas:

- Antiguo: el valor anterior para un cálculo de cambio de porcentaje o el importe total para un cálculo total/parcial.
- Nuevo: el valor nuevo para un cálculo de cambio de porcentaje o la parte del total para un cálculo total/parcial.
- Total: porcentaje total: $(\text{Nuevo} / \text{Antiguo}) * 100$
- Cambio: cambio de porcentaje: $((\text{Nuevo} / \text{Antiguo} / \text{Antiguo})) * 100100$

Introduzca la información en dos de los campos, mueva el cursor al valor que desee calcular y pulse **[[Soluc]]**.

1629 Vista de bono

ID 1629 · título localizado en HelpEs

El bono le permite calcular el precio o rédito de un bono.

- Fecha de liq.: Fecha de liquidación. El día en que la transferencia de efectivo o activos se ha completado y suele ser unos pocos días después de que se realizó la transacción comercial. Usa el formato AAAA.MMDD.
- Fecha Venc.: Fecha de vencimiento o fecha de rescate. Esta fecha siempre coincide con una fecha de cupón y es la fecha en que se canjea el bono. Usa el formato AAAA.MMDD.
- Cupón: tasa de cupón como un porcentaje anual. La tasa de cupón es la tasa de interés anual fija pagada por el emisor a un tenedor de bonos.
- Rescate: Valor de rescate. De forma predeterminada, es el precio de rescate el por valor nominal de 100.00. Al vencimiento, un bono tiene un valor de rescate del 100 % de su valor nominal
- Cal. 360: especifica que se debe utilizar un calendario de 30 días por mes, 360 días por año para los cálculos. El calendario de 360 días es útil para medir duraciones en los mercados financieros.
- Semestral: establece la frecuencia de pagos semestrales en lugar de anuales.

Introduzca toda la información conocida en todos los campos anteriores. Seleccione Rendimiento o Precio y pulse [[Soluc.]].

- Rendimiento: porcentaje de rendimiento hasta la fecha de vencimiento (o rescate) para un precio determinado.
- Precio: precio por valor nominal de 100.00 para un porcentaje de rendimiento dado.

Los siguientes campos de resultados se muestran al pulsar [[Soluc.]].

- Intereses devengados: intereses devengados desde el último cupón o fecha de pago hasta la fecha de liquidación para un rendimiento determinado.
- Duración modificada: medición de la sensibilidad del precio del bono con respecto a los cambios de rendimiento, que deriva de la duración de Macaulay.
- Duración de Macaulay: medición de la sensibilidad del precio del bono con respecto a los cambios de rendimiento.

1630 Vista de Black-Scholes

ID 1630 · título localizado en HelpEs

Black-Scholes es un modelo matemático útil para valorar las opciones europeas de adquisición y venta. Las opciones otorgan al tenedor el derecho de comprar o vender unidades de un activo subyacente durante un período de tiempo a un precio determinado. Una opción de rescate es el derecho a comprar y una opción de venta es el derecho a vender. En concreto, una opción de rescate le da al titular de la opción la posibilidad de comprar un número determinado de acciones de capital a un precio determinado antes de cierta fecha, independientemente del precio real de la acción en esa fecha. Una opción de venta le da al titular de la opción la posibilidad de vender un número determinado de acciones de capital a un precio determinado antes de una fecha determinada, independientemente del precio real de la acción en esa fecha.

Por ejemplo, supongamos que una opción de rescate permite la compra de 100 acciones de capital a 40,00 por acción por seis meses a partir de ahora. En ese punto de seis meses, si la acción vale 50,00, el titular de la opción puede comprarla por 40,00 y ganar 10,00 por acción de inmediato. Si la acción vale solamente 38.00 en ese punto de seis meses, la opción para comprar en 40.00 no se ejercería, pues perdería 2.00 por acción.

Los cálculos de Black-Scholes asumen una opción europea. Esto difiere de una opción americana en el sentido de que una opción europea solo se puede ejercer al final de su vida, o a su vencimiento. En igualdad de condiciones, el precio de una opción estadounidense suele ser superior al de una opción europea, ya que la opción estadounidense se puede negociar en cualquier momento hasta su vencimiento.

- Precio de la acción: valor actual del activo subyacente, también conocido como precio de contado.
- Precio de ejercicio: precio predeterminado a la que la opción acepta comprar o vender el activo subyacente al vencimiento.
- Fecha hasta el vencimiento: tiempo restante hasta el vencimiento de la opción en años.
- % libre de riesgo: tasa de interés libre de riesgo actual (por ejemplo, la tasa actual del bono del Tesoro de Estados Unidos).
- % Volatilidad: grado de cambio imprevisible del precio de la acción. Esto se suele aproximar a la desviación estándar de la variación del precio de la acción.
- % de dividendo: estimación del rendimiento medio del dividendo de la acción como porcentaje de su precio.

Introduzca los valores para todos los campos. Una vez que todos se han introducido, pulse [[Soluc.]] para calcular el precio de compra y el precio de venta.

- Precio de reclamo: valor de mercado justo estimado para una opción de reclamo en el vencimiento (una opción de reclamo es el derecho de comprar el activo e un precio dado).

- Precio por opción de venta: valor de mercado justo estimado para una opción de venta en el vencimiento (una opción de venta es el derecho de vender el activo a un precio dado).

1639 Vista numérica de Finanzas

ID 1639 · título localizado en HelpEs

La Vista numérica de Finanzas le permite elegir cuál cálculo financiero le gustaría realizar en la aplicación Finanzas. Entre ellos se incluyen:

- TVM (valor del dinero en el tiempo): usado para cálculos de interés compuesto que involucren flujos de efectivo regulares y uniformes
- Conversión de interés: hace la conversión entre tasas de interés nominal y efectiva
- Cálculo de fecha: calcula la diferencia entre dos fechas
- Flujo de efectivo: calcula el retorno a la inversión y el valor de los flujos de efectivo
- Depreciación: calcula la pérdida en el valor de los activos a lo largo del tiempo
- Umbral de rentabilidad: se usa para encontrar el umbral de rentabilidad entre la cantidad de unidades vendidas, los costos fijos, los costos de fabricación, el precio de venta y el lucro deseado
- Cambio del porcentaje: calcula un cambio basado en un porcentaje
- Bono: calcula el rédito o el precio de un bono
- Black-Scholes: usa la ecuación de Black-Scholes para valorar las inversiones

Primero, seleccione el cálculo que desea en el menú desplegable. Luego, presione Num para introducir la Vista numérica de Finanzas y solucione la ecuación elegida.

Para obtener una descripción más completa de cómo usar cada una de las opciones de finanzas, consulte el texto de ayuda del grupo de la Vista numérica de Finanzas.

0478 Variables de la aplicación Finanzas

ID 478 · título localizado en HelpEs

Además de las variables de modos (que son comunes a todas las aplicaciones), las variables de la aplicación Finanzas corresponden a los campos en la Vista numérica de la aplicación Finanzas .

1617 Variables simbólicas

ID 1617 · título localizado en HelpEs

Hay dos variables simbólicas de la aplicación Finanzas, cada una de las cuales corresponde a uno de los dos posibles campos en la Vista simbólica de la aplicación Finanzas:

- Método
- FinType

1618 Method — Determina el tipo de cálculo actual en la aplicación Finanzas.

ID 1618 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Method

Explicación española: Determina el tipo de cálculo actual en la aplicación Finanzas.

Method determina el tipo de cálculo actual en la aplicación Finanzas.

- Method := 0 para TVM
- Method := 1 para conversión de interés
- Method := 2 para cálculo de fecha
- Method := 3 para flujo de efectivo
- Method := 4 para depreciación
- Method := 5 para umbral de rentabilidad
- Method := 6 para cambio de porcentaje
- Method := 7 para bonos
- Method := 8 para Black-Scholes

1619 FinType — Determina el tipo de cálculo para la depreciación o los cálculos de porcentaje.

ID 1619 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: FinType

Explicación española: Determina el tipo de cálculo para la depreciación o los cálculos de porcentaje.

FinType determina el tipo de cálculo para la depreciación o los cálculos de porcentaje.

Con el Método=4 para la depreciación, los valores constantes y sus significados son los siguientes:

- FinType:= 0 para línea recta
- FinType:= 1 para los dígitos de la suma de los años
- FinType:= 2 para los saldos decrecientes
- FinType:= 3 para los saldos decrecientes cruzados
- FinType:= 4 para línea recta francesa
- FinType:= 5 para amortización francesa

Con el Método=6 para cálculos de porcentaje, los valores constantes y sus significados son los siguientes:

- FinType:= 0 para margen/incremento del precio
- FinType:= 1 por porcentaje/cambio

1541 Variables TVM

ID 1541 · título localizado en HelpEs

Después de que se realiza un cálculo de TVM (Valor tiempo del dinero) en la aplicación Finanzas, los valores se almacenan en las variables TVM.

0479 Variable de aplicación NbPmt

ID 479 · título recuperado de la página española

NbPmt - La cantidad de pagos de una inversión o préstamo.

n ► NbPmt, donde $n > 0$, establece el valor de NbPmt en n.

0480 Variable de aplicación IPYR

ID 480 · título recuperado de la página española

IPYR : La tasa de interés anual de una inversión o préstamo.

n ► IPYR establece el valor de IPYR a n.

0481 Variable de aplicación PV

ID 481 · título recuperado de la página española

PV : el valor presente de una inversión o préstamo.

n ► PV establece el valor de PV para n.

0482 Variable de aplicación PMT

ID 482 · título recuperado de la página española

PMT : el valor de un pago de un préstamo o inversión.

n ► PMT establece el valor de PMT en n.

0483 Variable de la aplicación FV

ID 483 · título recuperado de la página española

FV : El valor futuro de una inversión o un préstamo.

n ► FV establece el valor de FV en n.

0484 Variable de aplicación PPYR

ID 484 · título recuperado de la página española

PPYR : el número de pagos anuales para una inversión o préstamo.

n ► PPYR, donde $n > 0$, establece el valor de PPYR en n.

0485 Variable de aplicación CPYR

ID 485 · título recuperado de la página española

CPYR : El número de períodos capitalizables por año para una inversión o un préstamo. El valor predeterminado es 12.

n ► CPYR, donde $n > 0$, establece el valor de CPYR en n.

0486 Variable de aplicación BEG

ID 486 · título recuperado de la página española

BEG determina si el interés se capitaliza al principio o al final del período de capitalización.

0 ► BEG para interés compuesto al final del periodo (valor predeterminado)

1 ► BEG para interés compuesto al principio del periodo

0487 Variable de aplicación GSize

ID 487 · título recuperado de la página española

GSize : El tamaño de cada grupo para la tabla de amortización. El valor predeterminado es 12.

n ► GSize, donde $n > 0$, establece el valor de GSize en n.

1542 Variables de conversión de interés

ID 1542 · título localizado en HelpEs

Después de que se realiza un cálculo de Conversión de interés en la aplicación Finanzas, los valores se almacenan en las Variables de conversión de interés.

1544 NomInt — La tasa de interés nominal.

ID 1544 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: NomInt

Explicación española: La tasa de interés nominal.

NomInt - La tasa de interés nominal.

n ► NomInt , donde $0 \leq n \leq 100$, establece el valor de NomInt para n

1546 EffInt — La tasa de interés efectiva.

ID 1546 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: EffInt

Explicación española: La tasa de interés efectiva.

EffInt - La tasa de interés efectiva.

n ► EffInt , donde $0 \leq n \leq 100$, establece el valor de EffInt en n

1547 IntCPYR — El número de veces que los intereses se capitalizan por año

ID 1547 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: IntCPYR

Explicación española: El número de veces que los intereses se capitalizan por año

IntCPYR El número de veces que los intereses se capitalizan por año

n ► IntCPYR, donde $n > 0$, establece el valor de IntCPYR para n.

1593 Variables de cálculo de fechas

ID 1593 · título localizado en HelpEs

Después de que se realiza un cálculo de fecha, los datos se almacenan en las variables de cálculo de fechas.

1594 DateOne — La primera fecha utilizada en un cálculo de fecha.

ID 1594 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: DateOne

Explicación española: La primera fecha utilizada en un cálculo de fecha.

DateOne - La primera fecha utilizada en un cálculo de fecha. Utiliza el formato AAAA.MMDD.

n ► DateOne, donde n es AAAA.MMDD, establece el valor de DateOne en n

1595 DateTwo — La segunda fecha utilizada en un cálculo de fecha.

ID 1595 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: DateTwo

Explicación española: La segunda fecha utilizada en un cálculo de fecha.

DateTwo - La segunda fecha utilizada en un cálculo de fecha. Utiliza el formato AAAA.MMDD

n ► DateTwo, donde n es AAAA.MMDD, establece el valor de DateTwo en n

1596 DateDiff — La diferencia entre las dos fechas.

ID 1596 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: DateDiff

Explicación española: La diferencia entre las dos fechas.

DateDiff - La diferencia entre las dos fechas.

n ► DateDiff, donde n>0, establece el valor de DateDiff en n

1597 Date360 — Determina si se usa un año gregoriano estándar o de 360 días al realizar un cálculo de fecha.

ID 1597 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Date360

Explicación española: Determina si se usa un año gregoriano estándar o de 360 días al realizar un cálculo de fecha.

Date360 - Determina si se usa un año gregoriano estándar o de 360 días al realizar un cálculo de fecha.

0 ► Date360 para el año estándar de 365 días

1 ► Date360 para año de 360 días

1608 Variables de flujo de efectivo

ID 1608 · título localizado en HelpEs

Cuando se realizan los cálculos de flujo de efectivo, se almacenan los datos en las variables de flujo de efectivo.

1620 CFData — Proporciona acceso a la información de flujo de efectivo y es una lista de listas.

ID 1620 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CFData

Explicación española: Proporciona acceso a la información de flujo de efectivo y es una lista de listas.

Sintaxis

```
CFData
CFData(n)
CFData(n, opción)
CFData:={flujo de efectivo1, flujo de efectivo2, ... flujo de efectivoN}
CFData:=[flujo de efectivo1, flujo de efectivo2, ... flujo de efectivoN]
CFData:={{flujo de efectivo1, recuento1}, {flujo de efectivo2, recuento2}, ... {flujo de efectivoN, recuentoN}}
CFData:=[[flujo de efectivo1, recuento1], [flujo de efectivo2, recuento2], ... [flujo de efectivoN, recuentoN]]
CFData(n):=flujo de efectivo
CFData(n):={flujo de efectivo, count}
CFData(n):=[flujo de efectivo, count]
```

CFData proporciona acceso a la información de flujo de efectivo y es una lista de listas. Cada sublista contiene flujo de efectivo y recuento. El recuento por defecto es 1 si no se especifica.

CFData(n) hace referencia al par de flujo de efectivo y recuento numerado n. El flujo de efectivo inicial es el número 0.

CFData(n,opción) hace referencia al flujo de efectivo o al recuento del nth par, según el valor de opción. 1 es flujo de efectivo, 2 es el recuento.

Se puede almacenar una lista completa de listas o matriz que representa la información de flujo de efectivo en una sola operación.

1621 InvestInt — La tasa de interés de inversión del flujo de efectivo.

ID 1621 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: InvestInt

Explicación española: La tasa de interés de inversión del flujo de efectivo.

InvestInt - La tasa de interés de inversión del flujo de efectivo.

n ► InvestInt , donde $0 \leq n \leq 100$, establece el valor de InvestInt en n

1622 SafeInt — La tasa de interés segura del flujo de efectivo.

ID 1622 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: SafeInt

Explicación española: La tasa de interés segura del flujo de efectivo.

SafeInt - La tasa de interés segura del flujo de efectivo.

n ► SafeInt , donde $0 \leq n \leq 100$, establece el valor de SafeInt en n

1623 CFPYR — El número de flujos de efectivo por año.

ID 1623 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CFPYR

Explicación española: El número de flujos de efectivo por año.

CFPYR - El número de flujos de efectivo por año.

n ► CFPYR , donde $1 \leq n \leq 12$, establece el valor de CFPYR en n

1609 IRR — La tasa interna de retorno de un flujo de efectivo.

ID 1609 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: IRR

Explicación española: La tasa interna de retorno de un flujo de efectivo.

IRR - La tasa interna de retorno de un flujo de efectivo.

1610 MIRR — La tasa interna de retorno modificada de un flujo de efectivo.

ID 1610 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: MIRR

Explicación española: La tasa interna de retorno modificada de un flujo de efectivo.

MIRR - La tasa interna de retorno modificada de un flujo de efectivo.

1611 FMRR — Almacena la tasa de rendimiento de gestión financiera de un flujo de efectivo.

ID 1611 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: FMRR

Explicación española: Almacena la tasa de rendimiento de gestión financiera de un flujo de efectivo.

FMRR almacena la tasa de rendimiento de gestión financiera de un flujo de efectivo.

1612 TotalCF — El flujo de efectivo total.

ID 1612 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: TotalCF

Explicación española: El flujo de efectivo total.

TotalCF - El flujo de efectivo total.

1613 NPV — Valor actual neto de un flujo de efectivo.

ID 1613 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: NPV

Explicación española: Valor actual neto de un flujo de efectivo.

NPV - Valor actual neto de un flujo de efectivo.

1614 NFV — Valor futuro neto de un flujo de efectivo.

ID 1614 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: NFV

Explicación española: Valor futuro neto de un flujo de efectivo.

NFV - Valor futuro neto de un flujo de efectivo.

1615 NUS — Serie Uniforme Neta de un flujo de efectivo.

ID 1615 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: NUS

Explicación española: Serie Uniforme Neta de un flujo de efectivo.

NUS - Serie Uniforme Neta de un flujo de efectivo.

1662 RecuperaciónDes

ID 1662 · título localizado en HelpEs

DiscPayback - El período de recuperación descontada de un flujo de caja.

1616 Payback — El período de recuperación de un flujo de efectivo.

ID 1616 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Payback

Explicación española: El período de recuperación de un flujo de efectivo.

Payback - El período de recuperación de un flujo de efectivo.

1569 Variables de depreciación

ID 1569 · título localizado en HelpEs

Después de que se realiza un cálculo de depreciación en la aplicación Finanzas, los valores se almacenan en las Variables de depreciación.

1570 CostAsset — El costo depreciable de un activo en el momento de la compra.

ID 1570 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CostAsset

Explicación española: El costo depreciable de un activo en el momento de la compra.

CostAsset - El costo depreciable de un activo en el momento de la compra.

n ► CostAsset, donde $n > 0$, establece el valor de CostAsset en n

1571 SalvageAsset — La cantidad de dinero a la que se puede comprar o recuperar un activo al final de su vida.

ID 1571 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: SalvageAsset

Explicación española: La cantidad de dinero a la que se puede comprar o recuperar un activo al final de su vida.

SalvageAsset - La cantidad de dinero a la que se puede comprar o recuperar un activo al final de su vida.

n ► SalvageAsset, donde $n > 0$, establece el valor de SalvageAsset en n

1572 FirstAsset — El mes en que el activo se pone en servicio por primera vez.

ID 1572 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: FirstAsset

Explicación española: El mes en que el activo se pone en servicio por primera vez.

FirstAsset - El mes en que el activo se pone en servicio por primera vez. Normalmente, esto será 1. Una cantidad decimal indica un mes parcial.

n ► FirstAsset, donde $n \geq 1$, establece el valor de FirstAsset en n

1573 LifeAsset — La vida útil esperada de un producto.

ID 1573 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: LifeAsset

Explicación española: La vida útil esperada de un producto.

LifeAsset - La vida útil esperada de un producto.

n ► LifeAsset, donde $n \geq 1$, establece el valor de LifeAsset en n.

1574 FactorDepr — El factor de depreciación como porcentaje, utilizado con el método del saldo decreciente.

ID 1574 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: FactorDepr

Explicación española: El factor de depreciación como porcentaje, utilizado con el método del saldo decreciente.

FactorDepr - El factor de depreciación como porcentaje, utilizado con el método del saldo decreciente.

n ► FactorDepr, donde $n > 0$, establece el valor de FactorDepr en n

1661 FirstDateAsset — La fecha de primer uso para el estilo francés de Depreciación, que se ingresa como AAAA.MMDD

ID 1661 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: FirstDateAsset

Explicación española: La fecha de primer uso para el estilo francés de Depreciación, que se ingresa como AAAA.MMDD

FirstDateAsset - La fecha de primer uso para el estilo francés de Depreciación, que se ingresa como AAAA.MMDD
n ► FirstDateAsset, donde n es AAAA.MMDD, establece el valor de FirstDateAsset en n

1543 Variables de umbral de rentabilidad

ID 1543 · título localizado en HelpEs

Después de que se realiza un cálculo de umbral de rentabilidad en la aplicación de Finanzas, los valores se almacenan en las variables de umbral de rentabilidad.

1545 FixedCost — El costo fijo para desarrollar y comercializar un producto.

ID 1545 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: FixedCost

Explicación española: El costo fijo para desarrollar y comercializar un producto.

FixedCost - El costo fijo para desarrollar y comercializar un producto.

n ► FixedCost, donde $n > 0$, establece el valor de FixedCost en n.

1548 VariableCost — El costo de fabricación por unidad.

ID 1548 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: VariableCost

Explicación española: El costo de fabricación por unidad.

VariableCost - El costo de fabricación por unidad.

n ► VariableCost, donde $n > 0$, establece el valor de VariableCost en n

1549 SalePrice — El precio de venta por unidad.

ID 1549 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: SalePrice

Explicación española: El precio de venta por unidad.

SalePrice - El precio de venta por unidad.

n ► SalePrice, donde $n > 0$, establece el valor de SalePrice en n.

1550 Profit — La ganancia esperada.

ID 1550 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Profit

Explicación española: La ganancia esperada.

Profit - La ganancia esperada.

n ► Profit, donde $n > 0$, establece el valor de Profit en n.

1551 Quantity — El número de unidades vendidas.

ID 1551 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Quantity

Explicación española: El número de unidades vendidas.

Quantity - El número de unidades vendidas.

n ► Quantity, donde $n > 0$, establece el valor de Quantity en n.

1599 Variables de cambio de porcentaje

ID 1599 · título localizado en HelpEs

Después de que se realizan los cálculos de cambio de porcentaje, los datos se almacenan en las variables de cambio de porcentaje.

1600 Price — El precio de venta en los cálculos de incremento de precios.

ID 1600 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Price

Explicación española: El precio de venta en los cálculos de incremento de precios.

Price - El precio de venta en los cálculos de incremento de precios.

n ► Price establece el valor de Price en n

1601 Cost — El costo de un elemento en los cálculos de incremento de precios.

ID 1601 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Cost

Explicación española: El costo de un elemento en los cálculos de incremento de precios.

Cost - El costo de un elemento en los cálculos de incremento de precios.

n ► Cost establece el valor de Cost en n

1602 Margin — El margen en los cálculos de incremento de precios basado en el costo.

ID 1602 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Margin

Explicación española: El margen en los cálculos de incremento de precios basado en el costo.

Margin - El margen en los cálculos de incremento de precios basado en el costo.

n ► Margin establece el valor de Margin en n

1603 Markup — El porcentaje de incremento en los cálculos de incremento de precios.

ID 1603 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Markup

Explicación española: El porcentaje de incremento en los cálculos de incremento de precios.

Markup - El porcentaje de incremento en los cálculos de incremento de precios.

n ► Markup establece el valor de Markup en n

1604 OldValue — El valor anterior en cálculos de cambio porcentual y el total en cálculos total/parcial.

ID 1604 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: OldValue

Explicación española: El valor anterior en cálculos de cambio porcentual y el total en cálculos total/parcial.

OldValue - El valor anterior en cálculos de cambio porcentual y el total en cálculos total/parcial.

n ► OldValue establece el valor de OldValue en n

1605 NewValue — El valor nuevo en cálculos de cambio porcentual y el parcial en cálculos total/parcial.

ID 1605 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: NewValue

Explicación española: El valor nuevo en cálculos de cambio porcentual y el parcial en cálculos total/parcial.

NewValue - El valor nuevo en cálculos de cambio porcentual y el parcial en cálculos total/parcial.

n ► NewValue establece el valor de NewValue en n

1606 Total — El porcentaje del total en los cálculos total/parcial.

ID 1606 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Total

Explicación española: El porcentaje del total en los cálculos total/parcial.

Total - El porcentaje del total en los cálculos total/parcial.

n ► Total establece el valor de Total en n

1607 Change — El cambio porcentual en los cálculos de cambio de porcentaje.

ID 1607 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Change

Explicación española: El cambio porcentual en los cálculos de cambio de porcentaje.

Change - El cambio porcentual en los cálculos de cambio de porcentaje.

n ► Change establece el valor de Change en n

1575 Variables de bono

ID 1575 · título localizado en HelpEs

Después de que se realiza un cálculo de bono en la aplicación Finanzas, los valores se almacenan en las variables de bono.

1576 SetDate — La fecha de liquidación de un bono.

ID 1576 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: SetDate

Explicación española: La fecha de liquidación de un bono.

SetDate - La fecha de liquidación de un bono. Las fechas se deben introducir como AAAA.MMDD

n ► SetDate, donde n es YYYY.MMDD, establece el valor de SetDate en n

1577 MatDate — La fecha de vencimiento o la fecha de rescate de un bono.

ID 1577 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: MatDate

Explicación española: La fecha de vencimiento o la fecha de rescate de un bono.

MatDate - La fecha de vencimiento o la fecha de rescate de un bono. Las fechas se deben introducir como AAAA.MMDD

n ► MatDate, donde n es AAAA.MMDD, establece el valor de MatDate en n

1578 CpnPer — El porcentaje del cupón.

ID 1578 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CpnPer

Explicación española: El porcentaje del cupón.

CpnPer - El porcentaje del cupón.

n ► CpnPer establece el valor de CpnPer en n

1579 CallPrice — El precio o valor de rescate.

ID 1579 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CallPrice

Explicación española: El precio o valor de rescate.

CallPrice - El precio o valor de rescate.

n ► CallPrice establece el valor de CallPrice en n

1580 YieldBond — El porcentaje de rendimiento al vencimiento de un bono.

ID 1580 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: YieldBond

Explicación española: El porcentaje de rendimiento al vencimiento de un bono.

YieldBond - El porcentaje de rendimiento al vencimiento de un bono.

n ► YieldBond establece el valor de YieldBond en n

1581 PriceBond — El precio por valor de 100.00 de un bono.

ID 1581 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: PriceBond

Explicación española: El precio por valor de 100.00 de un bono.

PriceBond - El precio por valor de 100.00 de un bono.

n ► PriceBond establece el valor de PriceBond en n

1658 Accrued — El interés acumulado de un bono.

ID 1658 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Accrued

Explicación española: El interés acumulado de un bono.

Accrued - El interés acumulado de un bono.

n ► Accrued establece el valor de Accrued en n

1659 Modified — La duración modificada de un bono.

ID 1659 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Modified

Explicación española: La duración modificada de un bono.

Modified - La duración modificada de un bono.

n ► Modified establece el valor de Modified en n

1660 Macaulay — La duración Macaulay de un bono.

ID 1660 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Macaulay

Explicación española: La duración Macaulay de un bono.

Macaulay - La duración Macaulay de un bono.

n ► Macaulay establece el valor deMacaulay en n

1582 Bond360 — Determina si se usa un calendario gregoriano estándar o un calendario de 360 días .

ID 1582 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Bond360

Explicación española: Determina si se usa un calendario gregoriano estándar o un calendario de 360 días .

Bond360 - Determina si se usa un calendario gregoriano estándar o un calendario de 360 días .

0 ► Bond360 para el año estándar de 365 días

1 ► Bond360 para el año de 360 días

1583 SemiAnnual — Determina si los pagos se realizan anual o semestralmente.

ID 1583 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: SemiAnnual

Explicación española: Determina si los pagos se realizan anual o semestralmente.

SemiAnnual - Determina si los pagos se realizan anual o semestralmente.

0 ► SemiAnnual indica pagos anuales

1 ► SemiAnnual indica pagos semestrales

1584 Variables de Black-Scholes

ID 1584 · título localizado en HelpEs

Después de que se realiza un cálculo de Black-Scholes, los valores se almacenan en las variables de Black-Scholes.

1585 StockPrice — El precio de las acciones.

ID 1585 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: StockPrice

Explicación española: El precio de las acciones.

StockPrice - El precio de las acciones. Este es el precio actual del activo subyacente, también conocido como precio al contado.

n ► StockPrice establece el valor de StockPrice en n

1586 StrikePrice — El precio de ejecución.

ID 1586 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: StrikePrice

Explicación española: El precio de ejecución.

StrikePrice - El precio de ejecución. Este es el precio predeterminado al que la opción acuerda comprar o vender el activo subyacente al vencimiento, también conocido como precio de ejercicio.

n ► StrikePrice establece el valor de StrikePrice en n

1588 TimeMarket — El tiempo de comercialización de una opción.

ID 1588 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: TimeMarket

Explicación española: El tiempo de comercialización de una opción.

TimeMarket - El tiempo de comercialización de una opción.

n ► TimeMarket establece el valor de TimeMarket en n

1587 RiskFree — La tasa de interés libre de riesgo.

ID 1587 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: RiskFree

Explicación española: La tasa de interés libre de riesgo.

RiskFree - La tasa de interés libre de riesgo.

n ► RiskFree establece el valor de RiskFree en n

1589 Volatility — La volatilidad de un activo.

ID 1589 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Volatility

Explicación española: La volatilidad de un activo.

Volatility - La volatilidad de un activo.

n ► Volatility establece el valor de Volatility en n

1590 Dividend — El porcentaje de dividendo.

ID 1590 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Dividend

Explicación española: El porcentaje de dividendo.

Dividend - El porcentaje de dividendo.

n ► Dividend establece el valor de Dividend en n

1591 BSCall — El precio de rescate de una opción.

ID 1591 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: BSCall

Explicación española: El precio de rescate de una opción.

BSCall - El precio de rescate de una opción.

n ► BSCall establece el valor de BSCall en n

1592 BSput — El precio de venta de una opción.

ID 1592 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: BSput

Explicación española: El precio de venta de una opción.

BSput - El precio de venta de una opción.

n ► BSput establece el valor de BSput en n

0488 Funciones de la aplicación Finanzas

ID 488 · título localizado en HelpEs

Las funciones específicas de la aplicación Finanzas se enumeran en esta sección.

1663 Funciones de TVM

ID 1663 · título localizado en HelpEs

Las funciones específicas del valor tiempo del dinero se enumeran en esta sección.

0489 Función de la aplicación TvmNbPmt

Sintaxis

`TvmNbPmt(IPYR, PV, PMT, FV [, PPYR [, CPYR [, BEG]])`

Calcula el número de pagos en una inversión o un préstamo.

- IPYR: la tasa de interés anual
- PV: el valor actual de la inversión o préstamo
- PMTV: el valor del pago
- FV: el valor futuro de la inversión o el préstamo
- PPYR: el número de pagos anuales
- CPYR: el número de períodos de capitalización por año
- BEG: los pagos efectuados al principio (1) o al final (0) del período

Los argumentos PPYR, CPYR y BEG son opcionales; si no se suministra, PPYR= 12, CPYR=PPYR y BEG= 0.

0490 Función de la aplicación TvmIPYR**Sintaxis**

`TvmIPYR(NbPmt, PV, PMT, FV [, PPYR [, CPYR [, BEG]])`

Calcula la tasa de interés por año en una inversión o un préstamo.

- NbPmt: el número de pagos
- PV: el valor actual de la inversión o préstamo
- PMT: el valor del pago
- FV: el valor futuro de la inversión o el préstamo
- PPYR: el número de pagos anuales
- CPYR: el número de períodos de capitalización por año
- BEG: los pagos efectuados al principio (1) o al final (0) del período

Los argumentos PPYR, CPYR y BEG son opcionales; si no se suministra, PPYR= 12, CPYR=PPYR y BEG= 0.

0491 Función de la aplicación TvmPV**Sintaxis**

`TvmPV(NbPmt, IPYR, PMT, FV [, PPYR [, CPYR [, BEG]])`

Calcula el valor actual de una inversión o un préstamo.

- NbPmt: el número de pagos
- IPYR: la tasa de interés anual
- PMTV: el valor del pago
- FV: el valor futuro de la inversión o el préstamo
- PPYR: el número de pagos anuales
- CPYR: el número de períodos de capitalización por año
- BEG: los pagos efectuados al principio (1) o al final (0) del período

Los argumentos PPYR, CPYR y BEG son opcionales; si no se suministra, PPYR= 12, CPYR=PPYR y BEG= 0.

0492 Función de la aplicación TvmPMT**Sintaxis**

`TvmPMT(NbPmt, IPYR, PV, FV [, PPYR [, CPYR [, BEG]])`

Calcula el valor de un pago para una inversión o un préstamo.

- NbPmt: el número de pagos
- IPYR: la tasa de interés anual

- PV: el valor actual de la inversión o préstamo
- FV: el valor futuro de la inversión o el préstamo
- PPYR: el número de pagos anuales
- CPYR: el número de períodos de capitalización por año
- BEG: los pagos efectuados al principio (1) o al final (0) del período

Los argumentos PPYR, CPYR y BEG son opcionales; si no se suministra, PPYR= 12, CPYR=PPYR y BEG= 0.

0493 Función de la aplicación TvmFV

ID 493 · título recuperado de la página española

Sintaxis

TvmFV(NbPmt, IPYR, PV, PMT [, PPYR [, CPYR [, BEG]])

Calcula el valor futuro de una inversión o un préstamo

- NbPmt: el número de pagos
- IPYR: la tasa de interés anual
- PV: el valor actual de la inversión o préstamo
- PMT: el valor del pago
- PPYR: el número de pagos anuales
- CPYR: el número de períodos de capitalización por año
- BEG: los pagos efectuados al principio (1) o al final (0) del período

Los argumentos PPYR, CPYR y BEG son opcionales; si no se suministra, PPYR= 12, CPYR=PPYR y BEG= 0.

1664 Funciones de conversión de intereses

ID 1664 · título localizado en HelpEs

Las funciones específicas para la conversión de intereses se enumeran en esta sección.

1553 IntConvNom — Devuelve la tasa de interés nominal en un cálculo de Conversión de interés cuando se dan la...

ID 1553 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: IntConvNom

Explicación española: Devuelve la tasa de interés nominal en un cálculo de Conversión de interés cuando se dan la...

Sintaxis

IntConvNom(tasa_efectiva, compuestos_por_año)

IntConvNom devuelve la tasa de interés nominal en un cálculo de Conversión de interés cuando se dan la tasa_efectiva y la cantidad de compuestos_por_año.

1552 IntConvEff — Devuelve la tasa de interés efectivo en un cálculo de Conversión de interés cuando se dan la...

ID 1552 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: IntConvEff

Explicación española: Devuelve la tasa de interés efectivo en un cálculo de Conversión de interés cuando se dan la...

Sintaxis

IntConvEff(tasa_nominal, compuestos_por_año)

IntConvEff devuelve la tasa de interés efectivo en un cálculo de Conversión de interés cuando se dan la tasa_nominal y el número de compuestos_por_año.

1554 IntConvCPYR — Devuelve el número de periodos compuestos en un año en un cálculo de Conversión de interés cuando se dan...

ID 1554 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: IntConvCPYR

Explicación española: Devuelve el número de periodos compuestos en un año en un cálculo de Conversión de interés cuando se dan...

Sintaxis

IntConvCPYR(tasa_nominal, tasa_efectiva)

IntConvCPYR devuelve el número de periodos compuestos en un año en un cálculo de Conversión de interés cuando se dan la tasa_nominal y la tasa_efectiva.

1560 DateDays — Devuelve la diferencia entre dos días cuando se dan dos fechas (first_date y second_date como AAAA.MMDD).

ID 1560 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: DateDays

Explicación española: Devuelve la diferencia entre dos días cuando se dan dos fechas (first_date y second_date como AAAA.MMDD).

Sintaxis

DateDays(primer_fecha, segunda_fecha [, cal_360])

DateDays devuelve la diferencia entre dos días cuando se dan dos fechas (first_date y second_date como AAAA.MMDD). Opcionalmente, un 1 en el tercer campo, cal_360, especificará que se debe usar un calendario de 360 días (doce meses de 30 días).

1665 Funciones de flujo de efectivo

ID 1665 · título localizado en HelpEs

En esta sección se enumeran las funciones específicas al flujo de efectivo.

1523 CashFlowIRR — Devuelve la Tasa Interna de Retorno para datos del flujo de efectivo.

ID 1523 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CashFlowIRR

Explicación española: Devuelve la Tasa Interna de Retorno para datos del flujo de efectivo.

Sintaxis

CashFlowIRR(datos_del_flujo_de_efectivo [, flujos_de_efectivo_por_año])

CashFlowIRR devuelve la Tasa Interna de Retorno para datos del flujo de efectivo. flujos de efectivo por año especifica el número de flujos de efectivo por año. Si flujos de efectivo por año no se proporciona, se supone que es 1.

Introduzca datos del flujo de efectivo como una lista o matriz. Estos son ejemplos de formas de entrada válidas:

```
{cash_flow1, cash_flow2, ... cash_flowN}
[cash_flow1, cash_flow2, ... cash_flowN]
```

Si desea especificar el recuento de un flujo de efectivo, el flujo de efectivo debe venir primero seguido por el recuento. Si no especifica un recuento, se supondrá que es 1. Estos son ejemplos de formas de entrada válidas:

```
{cash_flow1, {cash_flow2, count2}, ... {cash_flowN, countN}}
[[cash_flow1, count1], cash_flow2, ... [cash_flowN, countN]]
```

1524 CashFlowMIRR — Devuelve la tasa interna de retorno modificada para datos del flujo de efectivo, tasa de inversión y...

ID 1524 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CashFlowMIRR

Explicación española: Devuelve la tasa interna de retorno modificada para datos del flujo de efectivo, tasa de inversión y...

Sintaxis

CashFlowMIRR(datos_del_flujo_de_efectivo, tasa_de_inversión, tasa_de_inversión_segura [, flujos_de_efectivo_por_año])

CashFlowMIRR devuelve la tasa interna de retorno modificada para datos del flujo de efectivo, tasa de inversión y tasa de inversión segura. Si flujos de efectivo por año no se proporciona, se supone que es 1.

Introduzca datos del flujo de efectivo como una lista o matriz. Para indicar que el mismo flujo de efectivo se repite más de una vez, ingrese el flujo de efectivo como lista o matriz con el flujo de efectivo seguido del recuento. Si no especifica un recuento, se supondrá que es 1. Estos son ejemplos de formularios de entrada válidos:

```
{cash_flow1, {cash_flow2, count2}, ... {cash_flowN, countN}}  
[[cash_flow1, count1], cash_flow2, ... [cash_flowN, countN]]
```

1525 CashFlowMIRR — Devuelve la tasa de rendimiento de la gestión financiera para datos del flujo de efectivo, tasa de...

ID 1525 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CashFlowMIRR

Explicación española: Devuelve la tasa de rendimiento de la gestión financiera para datos del flujo de efectivo, tasa de...

Sintaxis

CashFlowMIRR(datos_del_flujo_de_efectivo, tasa_de_inversión, tasa_de_inversión_segura [, flujos_de_efectivo_por_año])

CashFlowMIRR devuelve la tasa de rendimiento de la gestión financiera para datos del flujo de efectivo, tasa de inversión y tasa de inversión segura. Si flujos de efectivo por año no se proporciona, se supone que es 1.

Introduzca datos del flujo de efectivo como una lista o matriz. Para indicar que el mismo flujo de efectivo se repite más de una vez, ingrese el flujo de efectivo como lista o matriz con el flujo de efectivo seguido del recuento. Si no especifica un recuento, se supondrá que es 1. Estos son ejemplos de formularios de entrada válidos:

```
{cash_flow1, {cash_flow2, count2}, ... {cash_flowN, countN}}  
[[cash_flow1, count1], cash_flow2, ... [cash_flowN, countN]]
```

1526 CashFlowTotal — Calcula el total de todas las entradas para datos del flujo de efectivo.

ID 1526 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CashFlowTotal

Explicación española: Calcula el total de todas las entradas para datos del flujo de efectivo.

Sintaxis

CashFlowTotal(datos_del_flujo_de_efectivo)

CashFlowTotal calcula el total de todas las entradas para datos del flujo de efectivo.

Introduzca datos del flujo de efectivo como una lista o matriz. Para indicar que el mismo flujo de efectivo se repite más de una vez, ingrese el flujo de efectivo como lista o matriz con el flujo de efectivo seguido del recuento. Si no especifica un recuento, se supondrá que es 1. Estos son ejemplos de formularios de entrada válidos:

```
{cash_flow1, {cash_flow2, count2}, ... {cash_flowN, countN}}  
[[cash_flow1, count1], cash_flow2, ... [cash_flowN, countN]]
```

1527 CashFlowNPV — Calcula el valor actual neto para datos del flujo de efectivo y tasa de inversión.

ID 1527 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CashFlowNPV

Explicación española: Calcula el valor actual neto para datos del flujo de efectivo y tasa de inversión.

Sintaxis

CashFlowNPV(datos_del_flujo_de_efectivo, tasa_de_inversión [, flujos_de_efectivo_por_año])

CashFlowNPV calcula el valor actual neto para datos del flujo de efectivo y tasa de inversión. flujos de efectivo por año especifica el número de flujos de efectivo por año. Si flujos de efectivo por año no se proporciona, se supone que es 1.

Introduzca datos del flujo de efectivo como una lista o matriz. Para indicar que el mismo flujo de efectivo se repite más de una vez, ingrese el flujo de efectivo como lista o matriz con el flujo de efectivo seguido del recuento. Si no especifica un recuento, se supondrá que es 1. Estos son ejemplos de formularios de entrada válidos:

```
{cash_flow1, {cash_flow2, count2}, ... {cash_flowN, countN}}  
[[cash_flow1, count1], cash_flow2, ... [cash_flowN, countN]]
```

1528 CashFlowNFV — Calcula el valor futuro neto para datos del flujo de efectivo y tasa de inversión.

ID 1528 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CashFlowNFV

Explicación española: Calcula el valor futuro neto para datos del flujo de efectivo y tasa de inversión.

Sintaxis

CashFlowNFV(datos_del_flujo_de_efectivo, tasa_de_inversión [, flujos_de_efectivo_por_año])

CashFlowNFV calcula el valor futuro neto para datos del flujo de efectivo y tasa de inversión. flujos de efectivo por año especifica el número de flujos de efectivo por año. Si flujos de efectivo por año no se proporciona, se supone que es 1.

Introduzca datos del flujo de efectivo como una lista o matriz. Para indicar que el mismo flujo de efectivo se repite más de una vez, ingrese el flujo de efectivo como lista o matriz con el flujo de efectivo seguido del recuento. Si no especifica un recuento, se supondrá que es 1. Estos son ejemplos de formularios de entrada válidos:

```
{cash_flow1, {cash_flow2, count2}, ... {cash_flowN, countN}}  
[[cash_flow1, count1], cash_flow2, ... [cash_flowN, countN]]
```

1529 CashFlowNUS — Calcula la Serie Uniforme Neta para datos del flujo de efectivo y tasa de inversión.

ID 1529 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CashFlowNUS

Explicación española: Calcula la Serie Uniforme Neta para datos del flujo de efectivo y tasa de inversión.

Sintaxis

CashFlowNUS(datos_del_flujo_de_efectivo, tasa_de_inversión [, flujos_de_efectivo_por_año])

CashFlowNUS calcula la Serie Uniforme Neta para datos del flujo de efectivo y tasa de inversión. flujos de efectivo por año especifica el número de flujos de efectivo por año. Si flujos de efectivo por año no se proporciona, se supone que es 1.

Introduzca datos del flujo de efectivo como una lista o matriz. Para indicar que el mismo flujo de efectivo se repite más de una vez, ingrese el flujo de efectivo como lista o matriz con el flujo de efectivo seguido del recuento. Si no especifica un recuento, se supondrá que es 1. Estos son ejemplos de formularios de entrada válidos:

```
{cash_flow1, {cash_flow2, count2}, ... {cash_flowN, countN}}  
[[cash_flow1, count1], cash_flow2, ... [cash_flowN, countN]]
```

1530 CashFlowPB — Calcula el período de reembolso descontado para datos del flujo de efectivo y tasa de inversión.

ID 1530 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CashFlowPB

Explicación española: Calcula el período de reembolso descontado para datos del flujo de efectivo y tasa de inversión.

Sintaxis

CashFlowPB(datos_del_flujo_de_efectivo [, tasa_de_inversión])
CashFlowPB(datos_del_flujo_de_efectivo [, tasa_de_inversión, flujos_de_efectivo_por_año])

CashFlowPB calcula el período de reembolso descontado para datos del flujo de efectivo y tasa de inversión. tasa de inversión especifica la tasa de inversión para el descuento. Un valor de 0 para tasa de inversión calculará el reembolso sin descuento. flujos de efectivo por año especifica el número de flujos de efectivo por año. Si flujos de efectivo por año no se proporciona, se supone que es 1.

Introduzca datos del flujo de efectivo como una lista o matriz. Para indicar que el mismo flujo de efectivo se repite más de una vez, ingrese el flujo de efectivo como lista o matriz con el flujo de efectivo seguido del recuento. Si no

especifica un recuento, se supondrá que es 1. Estos son ejemplos de formularios de entrada válidos:

```
{cash_flow1, {cash_flow2, count2}, ... {cash_flowN, countN}}  
[[cash_flow1, count1], cash_flow2, ... [cash_flowN, countN]]
```

0496 Depreciate — Devuelve el cronograma de amortización cuando se da el método de cálculo, el costo depreciable en el...

ID 496 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Depreciate

Explicación española: Devuelve el cronograma de amortización cuando se da el método de cálculo, el costo depreciable en el...

Sintaxis

Depreciate(método, costo, rescate, duración, [primero], [factor])

Depreciate devuelve el cronograma de amortización cuando se da el método de cálculo, el costo depreciable en el momento de la compra, el monto de retorno esperado de la valor de rescate del activo, la duración esperada en años, el momento de primer uso y factor de depreciación como porcentaje.

El momento del primero uso se expresa como un número correspondiente al mes y la parte fraccional del mes (ejemplo: 2,5 = 1/2 del mes de febrero) para los métodos 0 a 3 y como fecha real (en formato aaaa.mmdd) para los métodos 4 y 5.

Para introducir el método use los siguientes números:

- 0: Línea recta
- 1: Suma de dígitos por año
- 2: Saldo decreciente
- 3: Saldo decreciente con cruce de línea recta
- 4: Línea recta sistema francés
- 5: Amortización sistema francés

El esquema de amortización se devuelve como una lista de listas, donde el número de la lista corresponde al año de depreciación.

```
{{Depreciation_Year_1,Depreciable_Value_Year_1,Book_Value_Year_1},  
{Depreciation_Year_2,Depreciable_Value_Year_2,Book_Value_Year_2},  
{Depreciation_Year_n,Depreciable_Value_Year_n,Book_Value_Year_n}}
```

 ...

1666 Funciones de umbral de rentabilidad

ID 1666 · título localizado en HelpEs

En esta sección se enumeran las funciones específicas del umbral de rentabilidad.

1555 BrkEvFixed — Devuelve el costo fijo para desarrollar y comercializar un producto en un cálculo de umbral de...

ID 1555 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: BrkEvFixed

Explicación española: Devuelve el costo fijo para desarrollar y comercializar un producto en un cálculo de umbral de...

Sintaxis

BrkEvFixed(cantidad, costo, precio, lucro)

BrkEvFixed devuelve el costo fijo para desarrollar y comercializar un producto en un cálculo de umbral de rentabilidad cuando se dan el costo per unidad vendida, el precio per unidad de ventas, el lucro esperado y la cantidad de unidades vendidas.

1556 BrkEvCost — Devuelve el costo por unidad en un cálculo de umbral de rentabilidad cuando se dan el costo_fijo de...

ID 1556 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: BrkEvCost

Explicación española: Devuelve el costo por unidad en un cálculo de umbral de rentabilidad cuando se dan el costo_fijo de...

Sintaxis

BrkEvCost(costo_fijo, cantidad, precio, lucro)

BrkEvCost devuelve el costo por unidad en un cálculo de umbral de rentabilidad cuando se dan el costo_fijo de marketing y desarrollo, el precio por unidad de ventas, el lucro esperado y la cantidad de unidades vendida.

1557 BrkEvPrice — Devuelve el precio unitario en un cálculo de umbral de rentabilidad cuando se dan el costo_fijo de...

ID 1557 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: BrkEvPrice

Explicación española: Devuelve el precio unitario en un cálculo de umbral de rentabilidad cuando se dan el costo_fijo de...

Sintaxis

BrkEvPrice(costo_fijo, cantidad, costo, lucro)

BrkEvPrice devuelve el precio unitario en un cálculo de umbral de rentabilidad cuando se dan el costo_fijo de marketing y desarrollo, el costo por unidad de fabricación, el lucro esperado y la cantidad de unidades vendidas.

1558 BrkEvProfit — Devuelve el lucro en un cálculo de umbral de rentabilidad cuando se dan el costo_fijo, el costo de...

ID 1558 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: BrkEvProfit

Explicación española: Devuelve el lucro en un cálculo de umbral de rentabilidad cuando se dan el costo_fijo, el costo de...

Sintaxis

BrkEvProfit(costo_fijo, cantidad, costo, precio)

BrkEvProfit devuelve el lucro en un cálculo de umbral de rentabilidad cuando se dan el costo_fijo, el costo de fabricación de cada unidad, el precio de cada unidad y la cantidad de unidades vendidas.

1559 BrkEvQuant — Devuelve la cantidad de unidades vendidas en un cálculo de umbral de rentabilidad cuando se dan el...

ID 1559 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: BrkEvQuant

Explicación española: Devuelve la cantidad de unidades vendidas en un cálculo de umbral de rentabilidad cuando se dan el...

Sintaxis

BrkEvQuant(costo_fijo, costo, precio, lucro)

BrkEvQuant devuelve la cantidad de unidades vendidas en un cálculo de umbral de rentabilidad cuando se dan el costo_fijo de marketing y desarrollo, el costo por unidad de fabricación, el precio por unidad de ventas y el lucro esperado.

1667 Funciones de cambio de porcentaje

ID 1667 · título localizado en HelpEs

En esta sección se enumeran las funciones específicas al cambio del porcentaje.

1561 ChangePrice — Devuelve el precio de venta de un artículo en un cálculo de márgenes cuando se le asigna el costo y...

ID 1561 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ChangePrice

Explicación española: Devuelve el precio de venta de un artículo en un cálculo de márgenes cuando se le asigna el costo y...

Sintaxis

ChangePrice(costo, porcentaje, opción)

ChangePrice devuelve el precio de venta de un artículo en un cálculo de márgenes cuando se le asigna el costo y porcentaje. Si el porcentaje es un margen, use 0 para opción. Para el porcentaje de margen, establezca opción en 1.

1562 ChangeCost — Devuelve el costo de un elemento en un cálculo de márgenes cuando se le asignan precio y porcentaje de...

ID 1562 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ChangeCost

Explicación española: Devuelve el costo de un elemento en un cálculo de márgenes cuando se le asignan precio y porcentaje de...

Sintaxis

ChangeCost(precio, porcentaje, opción)

ChangeCost devuelve el costo de un elemento en un cálculo de márgenes cuando se le asignan precio y porcentaje de ventas. Si el porcentaje es un incremento de precio, use 0 para opción. Para el porcentaje de margen, establezca opción en 1.

1563 PercentMargin — Devuelve el porcentaje de incremento del precio como porcentaje del costo o del margen en los cálculos...

ID 1563 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: PercentMargin

Explicación española: Devuelve el porcentaje de incremento del precio como porcentaje del costo o del margen en los cálculos...

Sintaxis

PercentMargin(costo, precio)

PercentMargin devuelve el porcentaje de incremento del precio como porcentaje del costo o del margen en los cálculos de incremento del precio cuando se dan el precio y el costo de ventas del elemento.

1564 PercentMarkup — Devuelve el incremento del precio como un porcentaje del precio en los cálculos de incremento del precio...

ID 1564 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: PercentMarkup

Explicación española: Devuelve el incremento del precio como un porcentaje del precio en los cálculos de incremento del precio...

Sintaxis

PercentMarkup(costo, precio)

PercentMarkup devuelve el incremento del precio como un porcentaje del precio en los cálculos de incremento del precio cuando se dan el precio y costo de ventas de un elemento.

1565 ChangeOld — Devuelve el número antiguo en un cálculo de cambio porcentual cuando se le asigna el número nuevo y...

ID 1565 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ChangeOld

Explicación española: Devuelve el número antiguo en un cálculo de cambio porcentual cuando se le asigna el número nuevo y...

Sintaxis

ChangeOld(nuevo, porcentaje, opción)

ChangeOld devuelve el número antiguo en un cálculo de cambio porcentual cuando se le asigna el número nuevo y porcentaje.

Cuando opción es 0, porcentaje es un valor de porcentaje total y ChangeOld usará un cálculo total parcial. (nuevo / (porcentaje / 100))

Cuando opción es 1, porcentaje es un valor de cambio porcentual y ChangeOld usará un cálculo de cambio de porcentaje. (nuevo / (1 + porcentaje / 100))

1566 ChangeNew — Devuelve el número nuevo en un cálculo de cambio de porcentaje cuando se le asigna el número antiguo y...

ID 1566 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ChangeNew

Explicación española: Devuelve el número nuevo en un cálculo de cambio de porcentaje cuando se le asigna el número antiguo y...

Sintaxis

ChangeNew(antiguo, porcentaje, opción)

ChangeNew devuelve el número nuevo en un cálculo de cambio de porcentaje cuando se le asigna el número antiguo y porcentaje,

Cuando opción es 0, porcentaje es un valor de porcentaje total y ChangeNew realizará un cálculo total parcial. (antiguo * (porcentaje/100))

Cuando opción es 1, porcentaje es un valor de cambio porcentual y ChangeNew realizará un cálculo de cambio de porcentaje. (antiguo * (1 + porcentaje/100))

1567 PercentTotal — Calcula el porcentaje total parcial cuando se le asignan números antiguo y nuevo.

ID 1567 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: PercentTotal

Explicación española: Calcula el porcentaje total parcial cuando se le asignan números antiguo y nuevo.

Sintaxis

PercentTotal(antiguo, nuevo)

PercentTotal calcula el porcentaje total parcial cuando se le asignan números antiguo y nuevo.

1568 PercentChange — Calcula el cambio porcentual cuando se le asignan números antiguo y nuevo.

ID 1568 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: PercentChange

Explicación española: Calcula el cambio porcentual cuando se le asignan números antiguo y nuevo.

Sintaxis

PercentChange(antiguo, nuevo)

PercentChange calcula el cambio porcentual cuando se le asignan números antiguo y nuevo.

1668 Funciones de bonos

ID 1668 · título localizado en HelpEs

En esta sección se enumeran las funciones específicas a los bonos.

0494 BondYield — Devuelve el rendimiento como porcentaje, calculado hasta el vencimiento o hasta la fecha de rescate.

ID 494 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: BondYield

Explicación española: Devuelve el rendimiento como porcentaje, calculado hasta el vencimiento o hasta la fecha de rescate.

Sintaxis

BondYield(settlement_date, maturity_date, price, coupon_percent, call_value, semi_annual, cal360)

Devuelve el rendimiento como porcentaje, calculado hasta el vencimiento o hasta la fecha de rescate.

Parámetros

- settlement_date: Fecha de liquidación en formato YYYY.MMDD.
- maturity_date: Fecha de vencimiento en formato YYYY.MMDD.
- price: Precio por cada 100 de valor nominal.
- coupon_percent: Tasa de cupón anual como porcentaje.
- call_value: Precio de rescate por cada 100 de valor nominal. Si es 0, la función calcula el rendimiento hasta el vencimiento; si es mayor que 0, calcula el rendimiento hasta el rescate.
- semi_annual: Frecuencia de pago de cupón. 0 para pagos anuales. 1 para pagos semestrales.
- cal360: Base de recuento de días. 0 para recuento real de días. 1 para un año de 360 días con meses de 30 días.

0495 BondPrice — Devuelve el precio por cada 100 de valor nominal, calculado hasta la fecha de vencimiento o la fecha de...

ID 495 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: BondPrice

Explicación española: Devuelve el precio por cada 100 de valor nominal, calculado hasta la fecha de vencimiento o la fecha de...

Sintaxis

BondPrice(settlement_date, maturity_date, coupon_percent, call_value, yield_percent, semi_annual, cal360)

Devuelve el precio por cada 100 de valor nominal, calculado hasta la fecha de vencimiento o la fecha de rescate.

Parámetros

- settlement_date: Fecha de liquidación en formato YYYY.MMDD.
- maturity_date: Fecha de vencimiento o fecha de rescate en formato YYYY.MMDD.
- coupon_percent: Tasa de cupón anual como porcentaje.
- call_value: Precio de rescate por cada 100 de valor nominal.
- yield_percent: Rendimiento como porcentaje.
- semi_annual: Frecuencia de pago de cupón. 0 para pagos anuales. 1 para pagos semestrales.
- cal360: Base de recuento de días. 0 para recuento real de días. 1 para un año de 360 días con meses de 30 días.

0497 BlackScholes — Devuelve una lista que contiene el precio de la opción de compra y el precio de la opción de venta.

ID 497 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: BlackScholes

Explicación española: Devuelve una lista que contiene el precio de la opción de compra y el precio de la opción de venta.

Sintaxis

BlackScholes(stock_price, strike_price, time_to_maturity, risk_free_interest_rate, stock_volatility, stock_dividend)

BlackScholes devuelve una lista que contiene el precio de la opción de compra y el precio de la opción de venta. El cálculo utiliza el precio actual de la acción subyacente (stock_price), el precio de ejercicio de la opción (strike_price), el tiempo hasta el vencimiento en años (time_to_maturity), el porcentaje de la tasa de interés libre

de riesgo (risk_free_interest_rate), el porcentaje de volatilidad de la acción (stock_volatility) y el porcentaje de dividendo de la acción (stock_dividend).

1670 Aplicación Explorador

ID 1670 · título localizado en HelpEs

La aplicación Explorador está diseñada para explorar las relaciones entre los parámetros de una función y la forma del gráfico de la función.

Hay dos maneras de explorar en la Vista de gráfico. Puede manipular un gráfico y observar los cambios correspondientes en su ecuación, o puede editar los parámetros en una ecuación y anotar los cambios correspondientes en su representación gráfica. La aplicación también tiene un modo de prueba. En el modo de prueba, puede probar su habilidad para hacer coincidir las ecuaciones con los gráficos.

Presione la tecla Apps para abrir la biblioteca de aplicaciones y, a continuación, seleccione Explorador.

1671 Vista simbólica del explorador

ID 1671 · título localizado en HelpEs

La aplicación se abre en la Vista Simbólica, donde se selecciona la familia de funciones que se desea explorar. La aplicación Explorador admite las siguientes familias de funciones:

- Lineal
- Cuadrática
- Cúbica
- Exponencial
- Logarítmica
- Sinusoidal

Pulse en el campo y pulse [[Elegir]] para seleccionar una familia de funciones para explorar.

1672 Vista de gráfico del explorador

ID 1672 · título localizado en HelpEs

La Vista de gráfico muestra una ecuación junto con su gráfico. La ecuación y el gráfico dependen de la elección de la familia de funciones que realizó en la Vista simbólica. Según la familia de funciones, la Vista de gráfico también puede mostrar valores numéricos asociados a la gráfica, como las intercepciones, etc. A menudo, hay varios tipos (o niveles) de ecuación disponibles para explorar. Puede elegir entre ellos pulsando la tecla de menú denominada [[Niv 1]] o [[Niv 2]], y así sucesivamente.

Pulse y arrastre el gráfico para trasladarlo. La ecuación se actualiza automáticamente. Abra o cierre los dedos sobre la pantalla para dilatarlo vertical u horizontalmente. Una vez más, la ecuación se actualiza automáticamente. El gráfico original se muestra punteado para fines de comparación. La forma de la ecuación se muestra en la parte superior derecha de la pantalla, con la ecuación actual que coincide con la gráfica justo debajo de ella. A medida que manipula el gráfico, la ecuación se actualiza para reflejar los cambios. También puede pulsar la ecuación y editar los parámetros de esta directamente. Presione Enter o pulse [[OK]] para ver la actualización del gráfico.

En la Vista de gráfico, las teclas de menú comunes son:

- [[Niv n]] permite cambiar entre diversas fórmulas de las familias de funciones seleccionadas
- [[Prueba]] entra en el modo de prueba

1715 Funciones de la aplicación Explorer

ID 1715 · título localizado en HelpEs

Las funciones específicas de la aplicación Explorer se enumeran en esta sección.

1716 Solucionar para pendiente

ID 1716 · título recuperado de la página española

Sintaxis

LinearSlope(x1, y1, x2, y2)

Soluc. para pendiente. Toma como entrada las coordenadas de dos puntos (x1,y1) and (x2,y2) y devuelve la pendiente de la línea que contiene los dos puntos.

1717 Soluc. para interceptación Y

ID 1717 · título recuperado de la página española

Sintaxis

LinearYIntercept(x, y, m)

Toma como entrada las coordenadas de un punto (x, y), y una pendiente m, y devuelve la interceptación y de la línea con la pendiente dada que contiene el punto dado.

1718 Solucionar ecuación cuadrática

ID 1718 · título recuperado de la página española

Sintaxis

QuadSolve(a, b, C)

Dados los coeficientes de una ecuación cuadrática $a \cdot x^2 + b \cdot x + c = 0$, devuelve las soluciones reales.

1719 Solucionar discriminante

ID 1719 · título recuperado de la página española

Sintaxis

QuadDelta(a, b, C)

Dados los coeficientes de una ecuación cuadrática $a \cdot x^2 + b \cdot x + c = 0$, devuelve el valor de la discriminante en la Fórmula cuadrática.

0505 Aplicación Paramétrica

ID 505 · título localizado en HelpEs

La aplicación Paramétrica le permite explorar la variación simultánea de dos variables, cada una de las cuales depende de un parámetro T. Los valores de estas dos ecuaciones cuando varía T se tratan como las coordenadas de x y de y de un punto que se muestra en el plano cartesiano. Estas ecuaciones se muestran en la Vista simbólica en el formulario $X=f(T)$ y $Y=g(T)$.

Una vez definido un par de ecuaciones paramétricas, se puede trazar el gráfico o examinar una tabla de valores para las ecuaciones.

Para iniciar la aplicación Paramétrica, vaya a la biblioteca de aplicaciones y pulse el icono de la aplicación Paramétrica. También puede utilizar la rueda para seleccionar el icono de la aplicación Paramétrica, luego pulse [[Inicio]] o presione Enter para iniciar la aplicación.

0506 Vista simbólica de Paramétrica

ID 506 · título localizado en HelpEs

La Vista simbólica Paramétrica contiene hasta diez definiciones paramétricas, cada una compuesta por dos ecuaciones: $X=f(T)$ e $Y=g(T)$.

Cada definición tiene un selector de color a la izquierda de su campo. Toque el cuadro de color para elegir el color de la gráfica correspondiente a esa definición.

Los botones del menú son:

- [[Editar]] abre un cuadro de entrada para editar la definición paramétrica seleccionada
- [[✓]] activa o desactiva la definición seleccionada para la representación gráfica y la creación de tablas
- [[T]] es una ayuda de escritura para introducir definiciones en términos de T
- [[Mostr.]] muestra la definición seleccionada en modo de pantalla completa con el desplazamiento horizontal y vertical habilitado
- [[Eval.]] resuelve las referencias a otras ecuaciones, como $X2(T)=X1(T)/5$

Resalte uno de los campos de definición y comience a introducir una expresión en T, o toque **[Editar]** para abrir una línea de edición y modificar una expresión existente.

0507 Vista de gráfico de Paramétrica

ID 507 · título localizado en HelpEs

Presione Plot para abrir la Vista de gráfico de Paramétrica. Esta vista muestra los gráficos de las ecuaciones paramétricas definidas y comprobadas en la Vista simbólica. La funcionalidad es la misma que la de la Vista de gráfico de Función, con la excepción de que las funciones **[Fcn]** no se aplican aquí. Toque **[Menu]** para abrir el menú.

Los botones del menú son:

- **[Zoom]** permite abrir el menú Zoom, con opciones para acercar o alejar el zoom, etc
- **[Trazar]** permite activar y desactivar el cursor de trazado
- **[Ir a]** permite colocar el cursor de trazado en el punto del gráfico con un valor determinado de T
- **[Defn]** muestra la definición simbólica del gráfico actual
- **[Menú]** permite activar y desactivar el menú

Utilice la rueda hacia la izquierda/derecha o pulse para realizar un trazado a lo largo de un gráfico. Utilice la rueda hacia arriba/abajo para cambiar de un gráfico a otro. Presione + para acercar la posición actual del cursor y presione - para alejarla. Ajuste el factor de zoom en el menú **[Zoom]**.

También puede utilizar todos los gestos comunes a la Vista de gráfico. Consulte Vista de gráfico para obtener más detalles.

0508 Configuración de gráfico de Paramétrica

ID 508 · título localizado en HelpEs

Pulse Shift Plot para acceder a la Config. de gráfico de Paramétrica. Esta vista permite controlar la apariencia de la ventana del gráfico, incluida la apariencia del cursor, si deben dibujarse o no los ejes, etc. Esta configuración tiene dos páginas.

En la primera página, los campos son:

- Rng T: el intervalo de valores para el parámetro T
- Incr T: el valor de incremento del parámetro T
- Rng X: el rango de creación de gráfico horizontal
- Rng Y: el rango de creación de gráfico vertical
- Mrc X: el espaciado entre marcas de graduación horizontal
- Mrc Y: el espaciado entre marcas de graduación vertical

Los botones del menú en la primera página son:

- **[Editar]** abre una línea de edición para editar el valor del campo seleccionado
- **[-Página 1/2 ▼-]** muestra la segunda página de la configuración

Pulse **[-Página 1/2 ▼-]** para ver la segunda página de la configuración. Aquí, los campos son:

- Ejes: activa o desactiva los ejes
- Etiquetas: activa o desactiva las etiquetas del eje
- Puntos cuadr.: activa o desactiva los puntos de la cuadrícula
- Líneas cuadríc.: activa o desactiva las líneas de la cuadrícula
- Cursor: permite elegir entre cursores estándar, invertido y parpadeante
- Método: permite elegir entre adaptable, Segmento de increm. fijo y Puntos de increm. fijo

Los botones del menú en la segunda página son:

- **[✓]** activa o desactiva la configuración actual
- **[Selec.]** permite elegir de un cuadro de elección
- **[-▲ Página 2/2-]** retorna a la primera página de la configuración

El campo Método requiere una explicación. De forma predeterminada, la calculadora HP Prime utiliza el método Adaptable, un método avanzado que ofrece resultados muy precisos. Puede seleccionar el método más tradicional, denominado Segm. de increm. fijo, que muestrea valores de X, calcula sus valores de Y correspondientes y, a continuación, traza los puntos a la vez que los conecta. O bien, puede seleccionar Ptos de increm. fijo, que actúa

igual que Segm. de increm. fijo pero no conecta los puntos.

0509 Vista numérica de Paramétrica

ID 509 · título localizado en HelpEs

Pulse Num para acceder a la Vista numérica de Paramétrica. La Vista numérica de Paramétrica se ha diseñado para crear y examinar tablas de valores X/Y/T, con base en las funciones definidas en la Vista simbólica.

Coloque la barra de resaltado en cualquier fila de la columna T, introduzca un valor real y toque [[OK]]. La configuración de la tabla cambiará. También puede acercar o alejar el zoom en cualquier fila de la tabla. Pulse + para acercar el zoom en una fila de la tabla y - para alejar el zoom.

Los botones del menú son:

- [[Zoom]] permite acercar o alejar el zoom en una fila de la tabla. Tenga en cuenta que en la Vista numérica, el zoom cambia el incremento entre los valores x consecutivos. Acercar el zoom disminuye el incremento; alejar el zoom aumenta el incremento. Los valores de la fila que acerque o aleje siguen siendo los mismos.
- [[Más]] abre un menú con las opciones de edición
- [[Ir a]] pasa a un valor determinado de la variable independiente
- [[Defn]] muestra la definición de la columna seleccionada

El menú [[Más]] incluye las siguientes opciones:

- Seleccionar Fila: selecciona la fila que contiene la celda seleccionada en ese momento; la fila se puede copiar y pegar en otros lugares Intercambia final: esta opción está disponible una vez hecha la selección de varias celdas. Intercambia las celdas del principio y el final de la selección actual. Incluir encabezados: lo mismo que Seleccionar fila, excepto que también se seleccionan los encabezados de la fila
- Selección: activa o desactiva el modo de selección
- Tamaño de la fuente: selecciona una fuente pequeña, mediana o de gran tamaño

También puede utilizar cualquiera de los gestos comunes a la Vista numérica. Consulte Vista numérica para obtener más detalles.

0510 Configuración numérica de Paramétrica

ID 510 · título localizado en HelpEs

Pulse Shift Num para acceder a la Configuración numérica de Paramétrica. Esta vista permite controlar la apariencia de la tabla en la Vista numérica, incluido qué valor de T se encuentra al principio de la tabla, cuál es el intervalo entre valores de T y cuál es el factor de zoom para acercar y alejar una fila de la tabla.

Los campos son los siguientes:

- Núm. inicial: primer valor de T que se muestra en la tabla
- Núm. increm.: valor de incremento de tabla para T
- Núm. Zoom: el factor de zoom para acercar o alejar
- Tipo de núm.: elegir entre tipos de tablas Automático: brinda los valores de T, X y de Y Generar propio: permite introducir los valores de T y la aplicación proporcionará los valores de X e Y correspondientes para cada definición verificada de la Vista simbólica

Los botones del menú son:

- [[Editar]] abre una línea de edición para editar el valor actual de un campo
- [[Selec.]] permite seleccionar un tipo de tabla
- [[Gráf→]] permite configurar Núm. inicial y Núm. increm. de modo que los valores de variable independiente de la tabla de la Vista numérica coincidan con los valores de la variable independiente al realizar trazados en la Vista de gráfico.

0511 Variables Paramétricas

ID 511 · título localizado en HelpEs

Para mostrar las variables relacionadas con la aplicación Paramétrica, pulse Vars, toque [[Apl.]] y seleccione Paramétrica.

La aplicación Paramétrica incluye variables en las categorías siguientes:

- Simbólica (consulte a continuación)

- Gráfico (consulte Variables de aplicaciones comunes)
- Numérica (consulte Variables de aplicaciones comunes)
- Modos (consulte Variables de aplicaciones comunes)

Variables de la Vista simbólica

Las variables de la aplicación Paramétrica son X0-X9 e Y0-Y9. Estas variables se definen siempre en pares y contienen expresiones algebraicas que dependen de la variable T.

'Xn := f(T)'

'Yn := g(T)'

Donde n representa un entero entre 0 y 9 (inclusive) y f(T) y g(T) son expresiones que dependen de T.

Ejemplos

X1 := '4*COS(6*T)'

Y1 := '4*SIN(T)'

0512 Aplicación Secuencia

ID 512 · título localizado en HelpEs

Las secuencias son expresiones que dependen de un parámetro de número entero $N > 0$. Las secuencias se pueden definir de manera explícita o de forma repetitiva en términos de uno o dos valores anteriores.

La aplicación Secuencia permite explorar hasta 10 secuencias, denominadas de U1 a U9.

Algunas de las secuencias más conocidas son las definiciones repetitivas de factorial:

$U1(1)=1$

$U1(N)=N*U1(N-1)$

y la secuencia Fibonacci:

$U1(1)=1$

$U1(2)=1$

$U1(N)=U1(N-1)+U1(N-2)$

Una vez definida una secuencia, se puede ver una tabla de sus valores o trazar su gráfico.

Para iniciar la aplicación Secuencia, vaya a la biblioteca de aplicaciones y pulse el icono de la aplicación Secuencia. También puede utilizar la rueda para seleccionar el icono de la aplicación Secuencia, luego pulse [[Inicio]] o presione Enter para iniciar la aplicación.

0513 Vista simbólica de Secuencia

ID 513 · título localizado en HelpEs

La aplicación Secuencia de HP Prime le permite definir secuencias de forma explícita o recursiva. Use esta vista para introducir y administrar hasta diez definiciones de secuencia. Las definiciones explícitas definen $U(N)$ en términos de N. Las definiciones recursivas hacia atrás pueden definir $U(N)$ en términos de $U(N-1)$ o tanto de $U(N-1)$ como de $U(N-2)$. De manera similar, las definiciones recursivas hacia adelante pueden definir $U(N+1)$ en términos de $U(N)$ o $U(N+2)$ en términos tanto de $U(N+1)$ como de $U(N+2)$. Finalmente, N puede empezar en 1 (el valor predeterminado), 0 o cualquier entero positivo < 1000 .

Los primeros dos campos en la Vista simbólica contienen los primeros dos valores numéricos en la secuencia. En el caso de una secuencia definida de forma explícita, ambos se pueden dejar en blanco. En el caso de una secuencia definida de forma recursiva, debe suministrar al menos uno de los dos, según la naturaleza de su definición. Observe que las etiquetas de estos valores cambian, según el valor inicial de N que eligió en el campo Opción.

El tercer campo es para su definición simbólica. El campo Opción contiene el valor inicial de N. Después de este campo hay una casilla de verificación. Si no se marca (la opción predeterminada) su definición simbólica es para $U(N)$. Si se marca la casilla, su definición simbólica es para $U(N+1)$ en caso de que haya introducido un único valor inicial para su secuencia, o $U(N+2)$ si introdujo dos valores numéricos para iniciar su secuencia.

Los botones del menú son:

- [[Editar]] permite abrir una línea de edición para editar la definición seleccionada
- [[✓]] permite activar o desactivar el elemento actual
- [[Selec.]] permite seleccionar el color del gráfico
- [[Mostr.]] muestra la definición de secuencia seleccionada en modo de pantalla completa con el desplazamiento horizontal y vertical activado
- [[Eval.]] permite resolver referencias a otras definiciones simbólicas, p. ej., $U2(N-1)+U1(N)$.

Cuando se introduce o edita la definición simbólica de una secuencia, los elementos adicionales del menú son:

- `[[N-2]]`, `[[N-1]]`, `[[N]]`, `[[N+1]]`, `[[U1]]` ayudas de escritura para introducir las definiciones de su secuencia
- `[[Canc.]]` permite cancelar la edición actual
- `[[OK]]` permite aceptar la edición actual

Resalte uno de los campos de definición y comience a introducir una expresión, o toque `[[Editar]]` para abrir una línea de edición y editar una expresión existente.

0514 Vista de gráfico de Secuencia

ID 514 · título localizado en HelpEs

Pulse Plot para abrir la Vista de gráfico y trabajar con gráficos de secuencia. Toque `[[Menú]]` para abrir el menú.

Los botones del menú son:

- `[[Zoom]]` permite abrir el menú Zoom, con opciones para acercar o alejar el zoom
- `[[Trazar]]` permite activar y desactivar el cursor de trazado
- `[[Ir a]]` permite colocar el cursor de trazado en el punto del gráfico con un valor determinado de n
- `[[Defn]]` muestra la definición simbólica de la secuencia actual
- `[[Menú]]` permite activar y desactivar el menú

Utilice la rueda hacia la izquierda/derecha o pulse para realizar un trazado a lo largo de una función. Utilice la rueda hacia arriba/abajo para cambiar de una función a otra. Presione + para acercar la posición actual del cursor y presione - para alejarla. Ajuste el factor de zoom en el menú `[[Zoom]]`.

También puede utilizar todos los gestos comunes a la Vista de gráfico. Consulte Vista de gráfico para obtener más detalles.

0515 Configuración de gráfico de Secuencia

ID 515 · título localizado en HelpEs

Pulse Shift Plot para acceder a la Config. de gráfico. Esta vista le permite configurar manualmente la ventana y la apariencia de los gráficos de secuencia. Esta configuración tiene dos páginas.

Los campos de la primera página son:

- Gráf. secuencia: permite seleccionar los gráficos escalonados o de tela de araña en cada secuencia
- Rng N: indica el rango de términos en los que realizar el gráfico para cada secuencia
- Rng X: el rango de creación de gráfico horizontal
- Rng Y: el rango de creación de gráfico vertical
- Mrc X: espaciado entre marcas de graduación horizontal
- Mrc Y: espaciado entre marcas de graduación vertical

Los botones del menú en la primera página son:

- `[[Selec.]]` permite abrir el cuadro de selección de Gráf. secuencia
- `[[Editar]]` permite abrir una línea de edición para editar el valor del campo seleccionado
- `[-Página 1/2 ▼-]` permite introducir la segunda página de la vista

Pulse `[-Página 1/2 ▼-]` para ver la segunda página de la configuración. Aquí, los campos son:

- Ejes: activa o desactiva los ejes
- Etiquetas: activa o desactiva las etiquetas del eje
- Puntos cuadr.: activa o desactiva los puntos de la cuadrícula
- Líneas cuadríc.: activa o desactiva las líneas de la cuadrícula
- Cursor: permite elegir entre cursores estándar, invertido y parpadeante

Los botones del menú en la segunda página son:

- `[[✓]]` activa o desactiva la configuración actual
- `[[Selec.]]` permite elegir de un cuadro de elección
- `[-▲ Página 2/2-]` retorna a la primera página de la configuración

0516 Vista numérica de Secuencia

Pulse Num para acceder a la Vista numérica. La Vista numérica se ha diseñado para crear y examinar tablas de valores de secuencia y términos, basándose en las funciones definidas en la Vista simbólica.

Los botones del menú son:

- **[[Zoom]]** permite acercar o alejar el zoom en una fila de la tabla. Tenga en cuenta que en la Vista numérica, el zoom cambia el incremento entre los valores x consecutivos. Acercar el zoom disminuye el incremento; alejar el zoom aumenta el incremento. Los valores de la fila que acerque o aleje siguen siendo los mismos.
- **[[Más]]** abre un menú con las opciones de edición
- **[[Ir a]]** pasa a un valor determinado de la variable independiente
- **[[Defn]]** muestra la definición de la columna seleccionada

El menú **[[Más]]** incluye las siguientes opciones:

- **Seleccionar Fila:** selecciona la fila que contiene la celda seleccionada en ese momento; la fila se puede copiar y pegar en otros lugares. Intercambia final: esta opción está disponible una vez hecha la selección de varias celdas. Intercambia las celdas del principio y el final de la selección actual. Incluir encabezados: lo mismo que Seleccionar fila, excepto que también se seleccionan los encabezados de la fila
- **Selección:** activa o desactiva el modo de selección
- **Tamaño de la fuente:** selecciona una fuente pequeña, mediana o de gran tamaño

Resalte cualquier valor en la columna N y escriba cualquier número cardinal mayor que o igual al valor de inicio para N. La configuración de la tabla cambiará para mostrar su valor.

También puede utilizar cualquiera de los gestos comunes a la Vista numérica. Consulte Vista numérica para obtener más detalles.

0517 Configuración numérica de Secuencia

Presione Shift Num para abrir la configuración numérica. Esta vista permite controlar la apariencia de la tabla en la Vista numérica, incluido qué valor de N se encuentra al principio de la tabla, cuál es el intervalo entre valores de N y cuál es el factor de zoom para acercar y alejar una fila de la tabla.

Núm. increm. debe ser un entero positivo (cualquier otro valor se omitirá).

Los campos son los siguientes:

- **Núm. inicial:** primer valor de N que se muestra en la tabla
- **Núm. increm.:** incremento entero positivo entre valores consecutivos de N
- **Núm. zoom:** factor de zoom del entero positivo para acercar y alejar
- **Tipo de núm.:** elegir entre tipos de tablas Automática: proporciona valores de N y de secuencia. Generar propio: permite que el usuario proporcione los valores de N y la aplicación proporcione los valores de secuencia correspondientes

Los botones del menú son:

- **[[Editar]]** abre una línea de edición para editar el valor actual de un campo
- **[[Selec.]]** permite seleccionar un tipo de tabla
- **[[Gráf→]]** permite configurar Núm. inicial y Núm. increm. de modo que los valores de variable independiente de la tabla de la Vista numérica coincidan con los valores de la variable independiente al realizar trazados en la Vista de gráfico.

0518 Variables de Secuencia

Para mostrar las variables relacionadas con la aplicación Secuencia, pulse Vars, toque **[[Apl.]]** y seleccione Secuencia.

La aplicación Secuencia tiene variables en las siguientes categorías:

- Simbólica (consulte a continuación)
- Gráfico (consulte Variables de aplicaciones comunes)
- Numérica (consulte Variables de aplicaciones comunes)
- Modos (consulte Variables de aplicaciones comunes)

Variables de la Vista simbólica

Las variables simbólicas de la aplicación Secuencia son U1 a U9 y U0. Estas variables contienen listas que definen una secuencia. El número de elementos de la lista depende del tipo de secuencia definido.

$U_x := \{ \text{'expression'} [, U_x(1), U_x(2) [, StartIndex [, Forward definition]]] \}$, donde x es un entero entre 0 y 9 inclusive y la expresión es una expresión algebraica en términos de cualquier combinación de lo siguiente:

- N
- $U_n(N-1)$
- $U_n(N-2)$

Ejemplos

```
U3 := {'N!'} define U3 como una secuencia de factoriales
U5 := {'U5(N-1)+1,1'} define U5 como la secuencia de números de recuento
U7 := {'U7(N-2)+U7(N-1),1,1'} define U7 como la secuencia Fibonacci
```

0519 Aplicación Polar

ID 519 · título localizado en HelpEs

La aplicación Polar permite explorar la representación gráfica de ecuaciones mediante coordenadas polares. Cada ecuación sigue la fórmula $R = f(\theta)$ y su representación gráfica es el conjunto de puntos cuyas coordenadas polares cumplen la relación $R = f(\theta)$.

Una vez definida una ecuación polar, se puede ver una tabla de sus valores o trazar su gráfico.

Para iniciar la aplicación Polar, vaya a la biblioteca de aplicaciones y pulse el icono de la aplicación Polar. También puede utilizar la rueda para seleccionar el icono de la aplicación Polar, luego pulse **[[Inicio]]** o presione Enter para iniciar la aplicación.

0520 Vista simbólica de Polar

ID 520 · título localizado en HelpEs

Pulse Symb para volver a esta vista en cualquier momento. La Vista simbólica de Polar contiene campos para definir hasta diez ecuaciones polares, cada una de las cuales define R en función de θ .

Cada ecuación tiene un selector de color a la izquierda de su campo. Toque el cuadro de color para elegir el color de la gráfica correspondiente a esa ecuación.

Los botones del menú son:

- **[[Editar]]** abre una línea de entrada para editar la definición polar seleccionada
- **[[✓]]** activa o desactiva la definición seleccionada para la representación gráfica y la creación de tablas
- **[[θ]]** es una ayuda de escritura para introducir definiciones en θ
- **[[Mostr.]]** muestra la definición seleccionada en modo de pantalla completa con el desplazamiento horizontal y vertical habilitado
- **[[Eval.]]** resuelve las referencias a otras ecuaciones polares, como $R_2(\theta) = 2 \cdot R_1(\theta)$

Resalte uno de los campos de definición y comience a introducir una expresión en θ , o toque **[[Editar]]** para abrir una línea de edición y modificar una expresión existente.

0521 Vista de gráfico de Polar

ID 521 · título localizado en HelpEs

Presione Plot para abrir la Vista de gráfico de Polar. Esta vista muestra los gráficos de las ecuaciones polares definidas en la Vista simbólica. La funcionalidad es la misma que la de la Vista de gráfico de Función, con la excepción de que las funciones Func. no se aplican aquí. Toque **[[Menú]]** para abrir el menú.

Los botones del menú son:

- **[[Zoom]]** permite abrir el menú Zoom, con opciones para acercar o alejar el zoom
- **[[Trazar]]** permite activar y desactivar el cursor de trazado
- **[[Ir a]]** permite colocar el cursor de trazado en el punto del gráfico con un valor determinado de θ .
- **[[Defn]]** muestra la definición simbólica del gráfico actual
- **[[Menú]]** permite activar y desactivar el menú

Utilice la rueda hacia la izquierda/derecha o pulse para realizar un trazado a lo largo de una función. Utilice la rueda hacia arriba/abajo para cambiar de una función a otra. Presione + para acercar la posición actual del cursor y presione - para alejarla. Ajuste el factor de zoom en el menú **[[Zoom]]**.

También puede utilizar todos los gestos comunes a la Vista de gráfico. Consulte Vista de gráfico para obtener más detalles.

0522 Configuración de gráfico de Polar

ID 522 · título localizado en HelpEs

Presione Shift Plot para acceder a la Config. de gráfico de Polar. Esta vista permite controlar la apariencia de la ventana del gráfico, incluida la apariencia del cursor, si deben dibujarse o no los ejes, etc. Esta configuración tiene dos páginas.

En la primera página, los campos son:

- Rng θ : intervalo de valores para la variable independiente θ
- Incr θ : valor de incremento para la variable independiente θ
- Rng X: el rango de creación de gráfico horizontal
- Rng Y: el rango de creación de gráfico vertical
- Mrc X: espaciado entre marcas de graduación horizontal
- Mrc Y: espaciado entre marcas de graduación vertical

Los botones del menú en la primera página son:

- **[[Editar]]** abre una línea de edición para editar el valor del campo seleccionado
- **[-Página 1/2 ▼-]** muestra la segunda página de la configuración

Pulse **[-Página 1/2 ▼-]** para ver la segunda página de la configuración. Aquí, los campos son:

- Ejes: activa o desactiva los ejes
- Etiquetas: activa o desactiva las etiquetas del eje
- Puntos cuadr.: activa o desactiva los puntos de la cuadrícula
- Líneas cuadríc.: activa o desactiva las líneas de la cuadrícula
- Cursor: permite elegir entre cursores estándar, invertido y parpadeante
- Método: permite elegir entre adaptable, Segmento de increm. fijo y Puntos de increm. fijo

Los botones del menú en la segunda página son:

- **[[✓]]** activa o desactiva la configuración actual
- **[[Selec.]]** permite elegir de un cuadro de elección
- **[-▲ Página 2/2-]** retorna a la primera página de la configuración

El campo Método requiere una explicación. De forma predeterminada, la HP Prime utiliza el método Adaptable, un método avanzado que da resultados muy precisos. Puede elegir el método más tradicional, denominado Segm. de increm. fijo, que muestra los valores x , calcula sus valores y correspondientes, y a continuación, traza los puntos y los conecta. O bien, puede seleccionar Puntos de increm. fijo, que funciona como Segm. de increm. fijo pero no conecta los puntos.

0523 Vista numérica de Polar

ID 523 · título localizado en HelpEs

Pulse Num para acceder a la Vista numérica de Polar. La Vista numérica de Polar se ha diseñado para crear y explorar una tabla de valores θ/R , basados en las definiciones de la Vista simbólica.

Coloque la barra de resaltado en cualquier fila de la columna T, introduzca un valor real y toque OK. La configuración de la tabla cambiará. También puede acercar o alejar el zoom en cualquier fila de la tabla. Pulse + para acercar el zoom en una fila de la tabla y - para alejar el zoom.

Los botones del menú son:

- **[[Zoom]]** permite acercar o alejar el zoom en una fila de la tabla. Tenga en cuenta que en la Vista numérica, el zoom cambia el incremento entre los valores x consecutivos. Acercar el zoom disminuye el incremento; alejar el zoom aumenta el incremento. Los valores de la fila que acerque o aleje siguen siendo los mismos.
- **[[Más]]** abre un menú con las opciones de edición
- **[[Ir a]]** pasa a un valor determinado de la variable independiente

- **[[Defn]]** muestra la definición de la columna seleccionada

El menú **[[Más]]** incluye las siguientes opciones:

- **Seleccionar Fila:** selecciona la fila que contiene la celda seleccionada en ese momento; la fila se puede copiar y pegar en otros lugares
- **Intercambia final:** esta opción está disponible una vez hecha la selección de varias celdas. Intercambia las celdas del principio y el final de la selección actual. Incluir encabezados: lo mismo que Seleccionar fila, excepto que también se seleccionan los encabezados de la fila
- **Selección:** activa o desactiva el modo de selección
- **Tamaño de la fuente:** selecciona una fuente pequeña, mediana o de gran tamaño

También puede utilizar cualquiera de los gestos comunes a la Vista numérica. Consulte Vista numérica para obtener más detalles.

0524 Configuración numérica de Polar

ID 524 · título localizado en HelpEs

Pulse Shift Num para acceder a la Configuración numérica de Polar. Esta vista permite controlar la apariencia de la tabla en la Vista numérica, incluido qué valor de θ se encuentra al principio de la tabla, cuál es el intervalo entre valores de θ y cuál es el factor de zoom para acercar y alejar una fila de la tabla.

Los campos son los siguientes:

- **Num Start:** el primer valor de θ mostrado en la tabla
- **Num Step:** la diferencia común entre valores θ consecutivos
- **Tipo de núm.:** elegir entre tipos de tablas Automática: proporciona valores de θ y R BuildYourOwn: usted suministra valores de θ ; la aplicación brinda los valores R correspondientes
- **Núm. Zoom:** el factor de zoom para acercar o alejar

Los botones del menú son:

- **[[Editar]]** permite abrir un cuadro de edición para editar el valor actual de un campo
- **[[Gráf→]]** permite configurar Núm. inicial y Núm. increm. de modo que los valores de variable independiente de la tabla de la Vista numérica coincidan con los valores de la variable independiente al realizar trazados en la Vista de gráfico.

0525 Variables de Polar

ID 525 · título localizado en HelpEs

Para visualizar las variables relativas a la aplicación Polar, pulse Vars, toque **[[Apl.]]** y seleccione Polar.

La aplicación Polar dispone de variables en las siguientes categorías:

- Simbólica (consulte a continuación)
- Gráfico (consulte Variables de aplicaciones comunes)
- Numérica (consulte Variables de aplicaciones comunes)
- Modos (consulte Variables de aplicaciones comunes)

Variables de la Vista simbólica

Las variables simbólicas de la aplicación Polar son R1 a R9 y R0. Estas variables contienen expresiones algebraicas en términos de θ .

$R_n = f(\theta)$, donde n corresponde a un entero entre 0 y 9 (inclusive) y $f(\theta)$ es una expresión algebraica en términos de θ .

1749 Aplicación Python

ID 1749 · título localizado en HelpEs

La aplicación Python le permite crear y usar programas y comandos de Python.

Python es un lenguaje de programación relativamente simple que se encuentra en muchas computadoras y se usa a menudo en la educación.

Por favor, visite <https://python.org> para obtener más información.

HP Prime usa MicroPython, que incluye un pequeño subconjunto de la biblioteca estándar completa de Python y está optimizado para funcionar en dispositivos con recursos limitados.

Por favor, visite <https://micropython.org> para obtener más detalles.

Se pueden encontrar tutoriales de programación en Python en muchos lugares en internet.

No se explicará el uso completo de Python en este sistema de ayuda.

Utilice la Vista Simbólica para ingresar y editar scripts de Python.

Utilice la Vista Numérica para interactuar directamente con el entorno de ejecución interactivo de Python.

1750 Vista Simbólica de Python

ID 1750 · título localizado en HelpEs

La vista simbólica de la aplicación Python es un editor de archivos de texto.

Presione en el elemento del menú `[[Más]]` `[[Archivos>]]` para ver la lista de todos los archivos asociados con esta aplicación y editarlos, agregar nuevos y/o eliminarlos.

Cada uno de estos archivos ahora puede ser utilizado como un archivo de Python, de la misma manera que un archivo en el directorio local puede ser utilizado en un programa de Python en una PC.

Estos archivos son archivos de aplicación y también se pueden acceder a través del comando AFiles.

1751 Plantillas de Python

ID 1751 · título localizado en HelpEs

El menú `[[Plant]]` proporciona plantillas para estructuras comunes de programas Python y le ayuda a insertarlas rápidamente en sus programas.

1752 Vista Numérica de Python

ID 1752 · título localizado en HelpEs

La vista Numérica proporciona una consola de Python para la interacción directa con el entorno de Python.

El entorno se inicializa al iniciar la aplicación y se borra al salir, eliminando todas las definiciones del usuario. Cambiar entre vistas dentro de la aplicación de Python preserva el estado del entorno, pero cambiar a otra aplicación provoca una reinicialización al regresar.

Para código persistente, use la vista Simbólica, donde las definiciones se conservan entre sesiones. Al entrar en la vista Numérica, el sistema puede ejecutar opcionalmente el código de la vista Simbólica de forma previa (configurable en Configuración de gráfico de Python).

El menú `[[Borrar]]` restablece el entorno. El menú `[[Cmds]]` proporciona acceso a módulos integrados con ayuda contextual disponible mediante la tecla Help.

1753 Configuración de gráfico de Python

ID 1753 · título localizado en HelpEs

La configuración de gráfico de Python se utiliza para configurar varios ajustes en el entorno de Python. Incluye configuraciones de configuración gráfica, asignación de recursos y comportamientos relacionados con la edición de código.

- X Rng y Y Rng: Estos campos definen los rangos para los ejes X y Y para los comandos gráficos que dependen de coordenadas cartesianas. Puedes leer y modificar estos ajustes de manera programática utilizando los comandos `get_cartesian` y `set_cartesian` del módulo `hprime`.
- Heap (kilobytes): Permite especificar la cantidad de memoria asignada al heap, expresada en kilobytes. El heap se utiliza para la asignación dinámica de memoria durante la ejecución del programa.
- Pila (bytes): Permite definir el tamaño de la pila nativa, medido en bytes. La pila se utiliza para almacenar información de llamadas a funciones, como direcciones de retorno, parámetros de función y datos de flujo de control durante la ejecución.
- Memoria disponible: Muestra la cantidad de memoria libre disponible en el dispositivo.
- Cargar archivos automáticamente cuando se cambian: Indica si los archivos deben recargarse automáticamente cuando se modifican. Cuando está habilitado, cualquier cambio realizado en los archivos se carga inmediatamente en el entorno sin intervención manual.
- Requerir confirmación: Al marcar, se asegura que se le pida al usuario confirmación cada vez que un archivo se recarga automáticamente debido a cambios.

1755 Módulos de Python

ID 1755 · título localizado en HelpEs

Esta parte del sistema de ayuda documenta todas las funciones integradas de Python.

2069 Módulo arit de Python

ID 2069 · título recuperado de la página española

El módulo arit proporciona una pequeña colección de funciones esenciales de teoría de números.

2070 isprime (arit) — Devuelve un booleano que indica si un entero `n` es primo.

ID 2070 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `isprime (arit)`

Explicación española: Devuelve un booleano que indica si un entero `n` es primo.

Sintaxis

`isprime(n)`

Devuelve un booleano que indica si un entero `n` es primo.

2071 nextprime (arit) — Devuelve el siguiente número primo mayor que un entero `n`.

ID 2071 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `nextprime (arit)`

Explicación española: Devuelve el siguiente número primo mayor que un entero `n`.

Sintaxis

`nextprime(n)`

Devuelve el siguiente número primo mayor que un entero `n`.

2072 prevprime (arit) — Devuelve el número primo anterior menor que un entero `n`.

ID 2072 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `prevprime (arit)`

Explicación española: Devuelve el número primo anterior menor que un entero `n`.

Sintaxis

`prevprime(n)`

Devuelve el número primo anterior menor que un entero `n`.

2073 ifactor (arit) — Devuelve la factorización de un entero `n` como un producto de potencias de primos.

ID 2073 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `ifactor (arit)`

Explicación española: Devuelve la factorización de un entero `n` como un producto de potencias de primos.

Sintaxis

`ifactor(n)`

Devuelve la factorización de un entero `n` como un producto de potencias de primos.

2074 gcd (arit) — Devuelve el máximo común divisor de los enteros `a` y `b`.

ID 2074 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `gcd (arit)`

Explicación española: Devuelve el máximo común divisor de los enteros ``a`` y ``b``.

Sintaxis

`gcd(a, b)`

Devuelve el máximo común divisor de los enteros `a` y `b`.

2075 lcm (arit) — Devuelve el mínimo común múltiplo de los enteros ``a`` y ``b``.

ID 2075 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `lcm (arit)`

Explicación española: Devuelve el mínimo común múltiplo de los enteros ``a`` y ``b``.

Sintaxis

`lcm(a, b)`

Devuelve el mínimo común múltiplo de los enteros `a` y `b`.

2076 iegcd (arit) — Devuelve el máximo común divisor extendido para los enteros ``a`` y ``b``.

ID 2076 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `iegcd (arit)`

Explicación española: Devuelve el máximo común divisor extendido para los enteros ``a`` y ``b``.

Sintaxis

`iegcd(a, b)`

Devuelve el máximo común divisor extendido para los enteros `a` y `b`.

Devuelve `[u, v, igcd(a, b)]` tal que $a*u + b*v = \text{igcd}(a, b)$

2077 char (arit) — Devuelve el carácter que representa el unicode especificado.

ID 2077 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `char (arit)`

Explicación española: Devuelve el carácter que representa el unicode especificado.

Sintaxis

`char(n)`

Devuelve el carácter que representa el unicode especificado.

Acepta listas o tuplas de enteros.

2078 asc (arit) — Devuelve una versión legible de cualquier objeto (cadenas, tuplas, listas, etc.).

ID 2078 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `asc (arit)`

Explicación española: Devuelve una versión legible de cualquier objeto (cadenas, tuplas, listas, etc.).

Sintaxis

`asc(n)`

Devuelve una versión legible de cualquier objeto (cadenas, tuplas, listas, etc.).

2079 euler (arit) — Devuelve la función ϕ de Euler evaluada en un entero ``n``.

ID 2079 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `euler (arit)`

Explicación española: Devuelve la función phi de Euler evaluada en un entero ``n``.

Sintaxis

`euler(n)`

Devuelve la función phi de Euler evaluada en un entero `n`.

Devuelve el número de enteros coprimos menores que `n`.

2080 nprimes (arit) — Devuelve el número de primos menores o iguales a un entero ``n``.

ID 2080 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `nprimes (arit)`

Explicación española: Devuelve el número de primos menores o iguales a un entero ``n``.

Sintaxis

`nprimes(n)`

Devuelve el número de primos menores o iguales a un entero `n`.

1978 Módulo array de Python

ID 1978 · título recuperado de la página española

El módulo `array` define un tipo de objeto que puede representar de manera compacta un arreglo de valores básicos: caracteres, enteros, números de punto flotante.

1979 array, clase (array)

ID 1979 · título localizado en HelpEs

Sintaxis

`array(código_de_tipo [, inicializador])`

Los arreglos son tipos de secuencia y se comportan de manera muy similar a las listas, excepto que el tipo de objetos almacenados en ellos está restringido. El tipo se especifica en el momento de la creación del objeto utilizando un código de tipo, que es un solo carácter.

A continuación se muestra una lista de tipos definidos en el formato:

código de tipo: tipo C, tipo Python, tamaño del elemento en bytes

'b': signed char, int, 1

'B': unsigned char, int, 1

'h': signed short, int, 2

'H': unsigned short, int, 2

'i': signed int, int, 2

'I': unsigned int, int, 2

'l': signed long, int, 4

'L': unsigned long, int, 4

'q': signed long long, int, 8

'Q': unsigned long long, int, 8

'f': float, float, 4

'd': double, float, 8

El inicializador (opcional) es una secuencia u objeto iterable (por ejemplo, lista, tupla) utilizado para inicializar el arreglo. Los valores en el inicializador deben ser consistentes con el tipo especificado por el código de tipo. Si no se proporciona un inicializador, se crea un arreglo vacío.

Ejemplo

```
import array
# Crear un arreglo de tipo 'B' (char sin signo)
# con un inicializador - una lista de enteros
arr = array.array('B', [1, 2, 3, 4, 5])
```

1980 append (array.array) — Agrega el elemento especificado `x` al final del arreglo.

ID 1980 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `append (array.array)`

Explicación española: Agrega el elemento especificado x al final del arreglo.

Sintaxis

`append(x)`

Agrega el elemento especificado x al final del arreglo. x debe coincidir con el código de tipo del arreglo. Esta operación modifica el arreglo en su lugar y no devuelve un valor.

Ejemplo

```
import array
arr = array.array('i', [1, 2, 3])
arr.append(4)
print(arr) # Salida: array('i', [1, 2, 3, 4])
```

1981 `extend (array.array)` — Extiende el arreglo agregando todos los elementos del iterable proporcionado.

ID 1981 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `extend (array.array)`

Explicación española: Extiende el arreglo agregando todos los elementos del iterable proporcionado.

Sintaxis

`extend(iterable)`

Extiende el arreglo agregando todos los elementos del iterable proporcionado. Cada elemento en el iterable debe coincidir con el código de tipo del arreglo. Esta operación modifica el arreglo en su lugar y no devuelve un valor.

Ejemplo

```
import array
arr = array.array('i', [1, 2, 3])
arr.extend([4, 5, 6])
print(arr) # Salida: array('i', [1, 2, 3, 4, 5, 6])
```

1756 Módulo `builtins` de Python

ID 1756 · título recuperado de la página española

El módulo `builtins` contiene las funciones, los tipos, las excepciones y las constantes integrados que están disponibles de forma predeterminada en MicroPython en la HP Prime. Normalmente, no es necesario importar este módulo explícitamente.

1757 `bool`, clase (`builtins`)

ID 1757 · título localizado en HelpEs

Sintaxis

`bool([x])`

Devuelve un valor booleano, ya sea `True` o `False`, construido a partir de un argumento dado x.

Si se omite x o es `None`, la función devuelve `False`. Si se pasa x, se convierte utilizando el procedimiento estándar de prueba de verdad de Python: la mayoría de los objetos se consideran verdaderos a menos que se definan explícitamente como falsos. Por ejemplo, los valores numéricos como 0 y 0.0, las secuencias vacías como [], "" y los mapeos vacíos como {} se consideran falsos. Todos los demás valores se consideran verdaderos.

La clase `bool` es una subclase de `int`, por lo que `True` es equivalente a 1 y `False` es equivalente a 0.

Ejemplos

```
bool(10) # True
bool([]) # False
```

1758 `bytes`, clase (`builtins`)

Sintaxis

```
bytes([source=b'' [, encoding='utf-8' [, errors='strict']]])
```

Devuelve un nuevo objeto bytes, que es una secuencia inmutable de enteros en el rango $0 \leq x < 256$. Los bytes son una versión inmutable de bytearray – tiene los mismos métodos no mutables y el mismo comportamiento de indexación y segmentación.

En consecuencia, los argumentos del constructor se interpretan como para bytearray().

Los objetos bytes también se pueden crear con literales, ver Literales de cadenas y bytes.

1778 center (builtins.bytes) — Devuelve una copia del objeto centrado en una secuencia de longitud `width`.

Referencia traducida

Identificador técnico: `center (builtins.bytes)`

Explicación española: Devuelve una copia del objeto centrado en una secuencia de longitud `width`.

Sintaxis

```
center(width)
```

Devuelve una copia del objeto centrado en una secuencia de longitud `width`. El relleno se realiza utilizando el carácter de espacio. Para objetos de bytes, se devuelve la secuencia original si `width` es menor o igual a `len(s)`.

1775 count (builtins.bytes) — Devuelve el número de ocurrencias no superpuestas de la subsecuencia `sub` en el rango `[start, end]`.

Referencia traducida

Identificador técnico: `count (builtins.bytes)`

Explicación española: Devuelve el número de ocurrencias no superpuestas de la subsecuencia `sub` en el rango `[start, end]`.

Sintaxis

```
count(sub [, start [, end]])
```

Devuelve el número de ocurrencias no superpuestas de la subsecuencia `sub` en el rango `[start, end]`. Los argumentos opcionales `start` y `end` se interpretan como en la notación de corte.

La subsecuencia a buscar puede ser cualquier objeto similar a bytes o un entero en el rango de 0 a 255.

1759 decode (builtins.bytes) — Devuelve una cadena decodificada de los bytes dados.

Referencia traducida

Identificador técnico: `decode (builtins.bytes)`

Explicación española: Devuelve una cadena decodificada de los bytes dados.

Sintaxis

```
decode(encoding="utf-8", errors="strict")
```

Devuelve una cadena decodificada de los bytes dados. La codificación predeterminada y la única compatible es 'utf-8'. El modo de manejo de errores predeterminado y el único compatible es 'strict', lo que significa que los errores de codificación generan un `UnicodeError`.

1769 endswith (builtins.bytes) — Devuelve `True` si los datos binarios terminan con el sufijo especificado, de lo contrario devuelve `False`.

Referencia traducida

Identificador técnico: `endswith (builtins.bytes)`

Explicación española: Devuelve `True` si los datos binarios terminan con el sufijo especificado, de lo contrario devuelve `False`.

Sintaxis

`endswith(suffix)`

Devuelve True si los datos binarios terminan con el sufijo especificado, de lo contrario devuelve False.

El sufijo a buscar puede ser cualquier objeto similar a bytes.

1760 find (builtins.bytes) — Devuelve el índice más bajo en los datos donde se encuentra la subsecuencia ``sub``, de modo que ``sub`...`

ID 1760 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `find (builtins.bytes)`

Explicación española: Devuelve el índice más bajo en los datos donde se encuentra la subsecuencia ``sub``, de modo que ``sub`...`

Sintaxis

`find(sub [, start [, end]])`

Devuelve el índice más bajo en los datos donde se encuentra la subsecuencia `sub`, de modo que `sub` esté contenida en la porción `s[start:end]`. Los argumentos opcionales `start` y `end` se interpretan como en la notación de porciones. Devuelve -1 si no se encuentra `sub`.

La subsecuencia a buscar puede ser cualquier objeto similar a bytes o un entero en el rango de 0 a 255.

1773 format (builtins.bytes) — Realiza una operación de formateo en un objeto bytes.

ID 1773 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `format (builtins.bytes)`

Explicación española: Realiza una operación de formateo en un objeto bytes.

Sintaxis

`format(*args, **kwargs)`

Realiza una operación de formateo en un objeto bytes. Los bytes en los que se llama este método pueden contener texto literal o campos de reemplazo delimitados por llaves `{}`. Cada campo de reemplazo contiene el índice numérico de un argumento posicional, o el nombre de un argumento de palabra clave. El resultado es una cadena (no bytes), donde cada campo de reemplazo es reemplazado con el valor de cadena del argumento correspondiente.

Para más detalles, consulte el método `format()` de `builtins.str`.

Ejemplos

```
b'La suma de 1 + 2 es {0}'.format(1+2)
# 'La suma de 1 + 2 es 3'
```

```
b'Hola, {nombre}!'.format(nombre='Juan')
# 'Hola, Juan!'
```

1762 index (builtins.bytes) — Similar a `find()`, pero lanza `ValueError` cuando no se encuentra la subsecuencia.

ID 1762 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `index (builtins.bytes)`

Explicación española: Similar a `find()`, pero lanza `ValueError` cuando no se encuentra la subsecuencia.

Sintaxis

`index(sub [, start [, end]])`

Similar a `find()`, pero lanza `ValueError` cuando no se encuentra la subsecuencia.

La subsecuencia a buscar puede ser cualquier objeto similar a bytes o un entero en el rango de 0 a 255.

1782 isalpha (builtins.bytes) — Devuelve True si todos los bytes en la secuencia son caracteres ASCII alfabéticos y la secuencia no está...

Referencia traducidaIdentificador técnico: `isalpha` (`builtins.bytes`)

Explicación española: Devuelve True si todos los bytes en la secuencia son caracteres ASCII alfabéticos y la secuencia no está...

Sintaxis`isalpha()`

Devuelve True si todos los bytes en la secuencia son caracteres ASCII alfabéticos y la secuencia no está vacía, de lo contrario devuelve False. Los caracteres ASCII alfabéticos son aquellos valores de byte en la secuencia `b'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ'`.

1783 isdigit (`builtins.bytes`) — Devuelve True si todos los bytes en la secuencia son dígitos decimales ASCII y la secuencia no está...

Referencia traducidaIdentificador técnico: `isdigit` (`builtins.bytes`)

Explicación española: Devuelve True si todos los bytes en la secuencia son dígitos decimales ASCII y la secuencia no está...

Sintaxis`isdigit()`

Devuelve True si todos los bytes en la secuencia son dígitos decimales ASCII y la secuencia no está vacía, de lo contrario, devuelve False. Los dígitos decimales ASCII son aquellos valores de byte en la secuencia `b'0123456789'`.

1785 islower (`builtins.bytes`) — Devuelve True si hay al menos un carácter ASCII en minúscula en la secuencia y ningún carácter ASCII en...

Referencia traducidaIdentificador técnico: `islower` (`builtins.bytes`)

Explicación española: Devuelve True si hay al menos un carácter ASCII en minúscula en la secuencia y ningún carácter ASCII en...

Sintaxis`islower()`

Devuelve True si hay al menos un carácter ASCII en minúscula en la secuencia y ningún carácter ASCII en mayúscula, de lo contrario False.

1781 isspace (`builtins.bytes`) — Devuelve True si todos los bytes en la secuencia son espacios en blanco ASCII y la secuencia no está...

Referencia traducidaIdentificador técnico: `isspace` (`builtins.bytes`)

Explicación española: Devuelve True si todos los bytes en la secuencia son espacios en blanco ASCII y la secuencia no está...

Sintaxis`isspace()`

Devuelve True si todos los bytes en la secuencia son espacios en blanco ASCII y la secuencia no está vacía, de lo contrario False. Los caracteres de espacio en blanco ASCII son aquellos valores de byte en la secuencia `b'\t\n\r\x0b\f'` (espacio, tabulador, nueva línea, retorno de carro, tabulador vertical, alimentación de formulario).

1784 isupper (`builtins.bytes`) — Devuelve True si hay al menos un carácter alfabético ASCII en mayúsculas en la secuencia y ningún...

Referencia traducida

Identificador técnico: `isupper` (`builtins.bytes`)

Explicación española: Devuelve True si hay al menos un carácter alfabético ASCII en mayúsculas en la secuencia y ningún...

Sintaxis

`isupper()`

Devuelve True si hay al menos un carácter alfabético ASCII en mayúsculas en la secuencia y ningún carácter ASCII en minúsculas, de lo contrario False.

1764 join (builtins.bytes) — Devuelve un objeto bytes o bytearray que es la concatenación de las secuencias de datos binarios en...

ID 1764 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `join` (`builtins.bytes`)

Explicación española: Devuelve un objeto bytes o bytearray que es la concatenación de las secuencias de datos binarios en...

Sintaxis

`join(iterable)`

Devuelve un objeto bytes o bytearray que es la concatenación de las secuencias de datos binarios en `iterable`. Se generará un `TypeError` si hay valores en `iterable` que no sean objetos de tipo bytes, incluidos los objetos `str`. El separador entre los elementos es el contenido del objeto bytes o bytearray que proporciona este método.

1779 lower (builtins.bytes) — Devuelve una copia de la secuencia con todos los caracteres ASCII en mayúsculas convertidos a su...

ID 1779 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `lower` (`builtins.bytes`)

Explicación española: Devuelve una copia de la secuencia con todos los caracteres ASCII en mayúsculas convertidos a su...

Sintaxis

`lower()`

Devuelve una copia de la secuencia con todos los caracteres ASCII en mayúsculas convertidos a su equivalente en minúsculas.

1771 lstrip (builtins.bytes) — Devuelve una copia de la secuencia con los bytes iniciales especificados eliminados.

ID 1771 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `lstrip` (`builtins.bytes`)

Explicación española: Devuelve una copia de la secuencia con los bytes iniciales especificados eliminados.

Sintaxis

`lstrip([chars])`

Devuelve una copia de la secuencia con los bytes iniciales especificados eliminados. El argumento `chars` es una secuencia binaria que especifica el conjunto de valores de bytes a eliminar; el nombre se refiere al hecho de que este método generalmente se usa con caracteres ASCII. Si se omite o es `None`, el argumento `chars` por defecto elimina los espacios en blanco ASCII. El argumento `chars` no es un prefijo; más bien, se eliminan todas las combinaciones de sus valores:

La secuencia binaria de valores de bytes a eliminar puede ser cualquier objeto similar a bytes.

1776 partition (builtins.bytes) — Divide la secuencia en la primera aparición de ``sep``, y devuelve una tupla de 3 elementos que contiene...

ID 1776 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `partition (builtins.bytes)`

Explicación española: Divide la secuencia en la primera aparición de ``sep``, y devuelve una tupla de 3 elementos que contiene...

Sintaxis

`partition(sep)`

Divide la secuencia en la primera aparición de `sep`, y devuelve una tupla de 3 elementos que contiene la parte antes del separador, el separador en sí o su copia en bytearray, y la parte después del separador. Si no se encuentra el separador, devuelve una tupla de 3 elementos que contiene una copia de la secuencia original, seguida de dos objetos bytes o bytearray vacíos.

El separador a buscar puede ser cualquier objeto similar a bytes.

1774 replace (builtins.bytes) — Devuelve una copia de la secuencia con todas las ocurrencias de la subsecuencia ``old`` reemplazadas por...

ID 1774 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `replace (builtins.bytes)`

Explicación española: Devuelve una copia de la secuencia con todas las ocurrencias de la subsecuencia ``old`` reemplazadas por...

Sintaxis

`replace(old, new [, count])`

Devuelve una copia de la secuencia con todas las ocurrencias de la subsecuencia `old` reemplazadas por `new`. Si se proporciona el argumento opcional `count`, solo se reemplazan las primeras `count` ocurrencias.

La subsecuencia a buscar y su reemplazo pueden ser cualquier objeto similar a bytes.

1761 rfind (builtins.bytes) — Devuelve el índice más alto en la secuencia donde se encuentra la subsecuencia `sub`, de modo que `sub` esté...

ID 1761 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `rfind (builtins.bytes)`

Explicación española: Devuelve el índice más alto en la secuencia donde se encuentra la subsecuencia `sub`, de modo que `sub` esté...

Sintaxis

`rfind(sub [, start [, end]])`

Devuelve el índice más alto en la secuencia donde se encuentra la subsecuencia `sub`, de modo que `sub` esté contenida dentro de `s[start:end]`. Los argumentos opcionales `start` y `end` se interpretan como en la notación de segmento. Devuelve -1 en caso de fallo.

La subsecuencia a buscar puede ser cualquier objeto similar a bytes o un entero en el rango de 0 a 255.

1763 rindex (builtins.bytes) — Similar a `rfind()` pero lanza un `ValueError` cuando no se encuentra la subsecuencia `sub`.

ID 1763 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `rindex (builtins.bytes)`

Explicación española: Similar a `rfind()` pero lanza un `ValueError` cuando no se encuentra la subsecuencia `sub`.

Sintaxis

`rindex(sub [, start [, end]])`

Similar a `rfind()` pero lanza un `ValueError` cuando no se encuentra la subsecuencia `sub`.

La subsecuencia a buscar puede ser cualquier objeto similar a bytes o un entero en el rango de 0 a 255.

1777 rpartition (builtins.bytes) — Divide la secuencia en la última aparición de ``sep`` y devuelve una tupla de 3 elementos que contiene la...

Referencia traducidaIdentificador técnico: `rpartition (builtins.bytes)`Explicación española: Divide la secuencia en la última aparición de ``sep`` y devuelve una tupla de 3 elementos que contiene la...**Sintaxis**`rpartition(sep)`

Divide la secuencia en la última aparición de `sep` y devuelve una tupla de 3 elementos que contiene la parte antes del separador, el separador en sí o su copia en `bytearray`, y la parte después del separador. Si no se encuentra el separador, devuelve una tupla de 3 elementos que contiene dos objetos `bytes` o `bytearray` vacíos, seguidos de una copia de la secuencia original.

El separador a buscar puede ser cualquier objeto similar a `bytes`.

1767 `rsplit (builtins.bytes)` — Divide la secuencia binaria en subsecuencias del mismo tipo, usando `sep` como la cadena delimitadora.

Referencia traducidaIdentificador técnico: `rsplit (builtins.bytes)`Explicación española: Divide la secuencia binaria en subsecuencias del mismo tipo, usando `sep` como la cadena delimitadora.**Sintaxis**`rsplit([sep=None [, maxsplit=-1]])`

Divide la secuencia binaria en subsecuencias del mismo tipo, usando `sep` como la cadena delimitadora. Si se da `maxsplit`, se realizan como máximo `maxsplit` divisiones, las más a la derecha. Si no se especifica `sep` o es `None`, cualquier subsecuencia que consista únicamente en espacios en blanco ASCII es un separador. Excepto por dividir desde la derecha, `rsplit()` se comporta como `split()`.

1772 `rstrip (builtins.bytes)` — Devuelve una copia de la secuencia con los bytes finales especificados eliminados.

Referencia traducidaIdentificador técnico: `rstrip (builtins.bytes)`

Explicación española: Devuelve una copia de la secuencia con los bytes finales especificados eliminados.

Sintaxis`rstrip([chars])`

Devuelve una copia de la secuencia con los bytes finales especificados eliminados. El argumento `chars` es una secuencia binaria que especifica el conjunto de valores de bytes a eliminar; el nombre se refiere al hecho de que este método se usa generalmente con caracteres ASCII. Si se omite o es `None`, el argumento `chars` por defecto elimina los espacios en blanco ASCII. El argumento `chars` no es un sufijo; más bien, se eliminan todas las combinaciones de sus valores:

La secuencia binaria de valores de bytes a eliminar puede ser cualquier objeto similar a `bytes`.

1765 `split (builtins.bytes)` — Divide la secuencia binaria en subsecuencias del mismo tipo, usando ``sep`` como la cadena delimitadora.

Referencia traducidaIdentificador técnico: `split (builtins.bytes)`Explicación española: Divide la secuencia binaria en subsecuencias del mismo tipo, usando ``sep`` como la cadena delimitadora.**Sintaxis**`split([sep=None [, maxsplit=-1]])`

Divide la secuencia binaria en subsecuencias del mismo tipo, usando `sep` como la cadena delimitadora. Si se da `maxsplit` y es no negativo, se realizan como máximo `maxsplit` divisiones (por lo tanto, la lista tendrá como

máximo `maxsplit+1` elementos). Si `maxsplit` no se especifica o es `-1`, no hay límite en el número de divisiones (se realizan todas las divisiones posibles).

Si se da `sep`, los delimitadores consecutivos no se agrupan y se consideran como delimitadores de subsecuencias vacías (por ejemplo, `b'1,,2'.split(b',')` devuelve `[b'1', b'', b'2']`). El argumento `sep` puede consistir en una secuencia de varios bytes (por ejemplo, `b'1<>2<>3'.split(b'<>')` devuelve `[b'1', b'2', b'3']`). Dividir una secuencia vacía con un separador especificado devuelve `[b'']` o `[bytearray(b'')]`, dependiendo del tipo de objeto que se esté dividiendo. El argumento `sep` puede ser cualquier objeto similar a bytes.

Si `sep` no se especifica o es `None`, se aplica un algoritmo de división diferente: las series de espacios en blanco ASCII consecutivos se consideran como un solo separador, y el resultado no contendrá cadenas vacías al principio o al final si la secuencia tiene espacios en blanco al principio o al final. En consecuencia, dividir una secuencia vacía o una secuencia que consiste únicamente en espacios en blanco ASCII sin un separador especificado devuelve `[]`.

1766 `splitlines` (`builtins.bytes`) — Devuelve una lista de las líneas en la secuencia binaria, dividiéndolas en los límites de línea ASCII.

ID 1766 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `splitlines` (`builtins.bytes`)

Explicación española: Devuelve una lista de las líneas en la secuencia binaria, dividiéndolas en los límites de línea ASCII.

Sintaxis

`splitlines([keepends=False])`

Devuelve una lista de las líneas en la secuencia binaria, dividiéndolas en los límites de línea ASCII. Este método utiliza el enfoque de nuevas líneas universales para dividir las líneas. Los saltos de línea no se incluyen en la lista resultante a menos que se proporcione `keepends` y sea verdadero.

1768 `startswith` (`builtins.bytes`) — Devuelve `True` si los datos binarios comienzan con el prefijo especificado, de lo contrario devuelve...

ID 1768 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `startswith` (`builtins.bytes`)

Explicación española: Devuelve `True` si los datos binarios comienzan con el prefijo especificado, de lo contrario devuelve...

Sintaxis

`startswith(prefix [, start])`

Devuelve `True` si los datos binarios comienzan con el prefijo especificado, de lo contrario devuelve `False`. Con el parámetro opcional `start`, la prueba comienza en esa posición.

El prefijo a buscar puede ser cualquier objeto similar a bytes.

1770 `strip` (`builtins.bytes`) — Devuelve una copia de la secuencia con los bytes iniciales y finales especificados eliminados.

ID 1770 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `strip` (`builtins.bytes`)

Explicación española: Devuelve una copia de la secuencia con los bytes iniciales y finales especificados eliminados.

Sintaxis

`strip([chars])`

Devuelve una copia de la secuencia con los bytes iniciales y finales especificados eliminados. El argumento `chars` es una secuencia binaria que especifica el conjunto de valores de bytes que se deben eliminar: el nombre se refiere al hecho de que este método se usa generalmente con caracteres ASCII. Si se omite o es `None`, el argumento `chars` elimina de forma predeterminada los caracteres de espacio en blanco ASCII. El argumento `chars` no es un prefijo ni un sufijo; más bien, se eliminan todas las combinaciones de sus valores.

La secuencia binaria de valores de bytes que se deben eliminar puede ser cualquier objeto similar a bytes.

Los siguientes métodos en objetos bytes y bytearray presuponen el uso de formatos binarios compatibles con ASCII y no deben aplicarse a datos binarios arbitrarios. Tenga en cuenta que todos los métodos de bytearray de esta sección no operan in-place, sino que producen nuevos objetos.

1780 upper (builtins.bytes) — Devuelve una copia de la secuencia con todos los caracteres ASCII en minúsculas convertidos a su...

ID 1780 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: upper (builtins.bytes)

Explicación española: Devuelve una copia de la secuencia con todos los caracteres ASCII en minúsculas convertidos a su...

Sintaxis

upper()

Devuelve una copia de la secuencia con todos los caracteres ASCII en minúsculas convertidos a su equivalente en mayúsculas.

1786 bytearray, clase (builtins)

ID 1786 · título localizado en HelpEs

Sintaxis

bytearray([entrada])

Devuelve un nuevo arreglo de bytes. La clase bytearray es una secuencia mutable de enteros en el rango $0 \leq x < 256$. Esta implementación de bytearray admite métodos básicos para agregar y extender su contenido.

El parámetro entrada es opcional y se puede usar para inicializar el arreglo de las siguientes maneras:

- Entero: Si se pasa un entero, define el tamaño del arreglo y lo inicializa con bytes nulos (`\x00`).
- Iterable: Si se pasa un iterable de enteros, cada entero debe estar en el rango $0 \leq x < 256$ y se utilizará como contenido inicial del arreglo.
- Objeto similar a bytes: Si se pasa un objeto similar a bytes (como un objeto bytes o `array.array`), se utiliza para inicializar el bytearray.
- Cadena: Si se pasa una cadena, se convierte automáticamente en bytes (este comportamiento difiere de la implementación estándar de Python, que requiere especificar la codificación).

Sin ningún argumento, se crea un bytearray vacío de tamaño 0.

Ejemplo

```
data = bytearray([65, 66, 67])
print(data) # Salida: bytearray(b'ABC')
```

1787 append (builtins.bytearray) — Agrega un solo byte b (un entero entre 0 y 255) al final del bytearray.

ID 1787 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: append (builtins.bytearray)

Explicación española: Agrega un solo byte b (un entero entre 0 y 255) al final del bytearray.

Sintaxis

append(b)

Agrega un solo byte b (un entero entre 0 y 255) al final del bytearray. Esta es una operación en el lugar que modifica el bytearray original.

Ejemplo

```
data = bytearray([1, 2, 3])
data.append(255)
print(data) # Salida: bytearray(b'\x01\x02\x03\xff')
```

1788 extend (builtins.bytearray) — Extiende el bytearray al agregar el contenido de ``input``.

ID 1788 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `extend (builtins bytearray)`

Explicación española: Extiende el bytearray al agregar el contenido de ``input``.

Sintaxis

```
extend(input)
```

Extiende el bytearray al agregar el contenido de `input`. `input` debe ser un objeto que soporte el protocolo de buffer, como `bytearray`, `bytes`, `array.array` u otros objetos compatibles con el buffer. Esta operación se realiza en su lugar y modifica el bytearray original.

Ejemplo

```
dt = bytearray(b'ABC')
dt.extend(b'D')
print(dt) # Salida: bytearray(b'ABCD')
```

1789 complex, clase (builtins)

ID 1789 · título localizado en HelpEs

Sintaxis

```
complex([real [, imag]])
```

Devuelve un número complejo con el valor `real + imag*1j` o convierte una cadena o número en un número complejo. Si el primer parámetro es una cadena, se interpretará como un número complejo y la función debe ser llamada sin un segundo parámetro. El segundo parámetro nunca puede ser una cadena. Cada argumento puede ser de cualquier tipo numérico (incluyendo complejo). Si se omite `imag`, por defecto es cero y el constructor sirve como una conversión numérica como `int` y `float`. Si se omiten ambos argumentos, devuelve `0j`.

1790 dict, clase (builtins)

ID 1790 · título localizado en HelpEs

Sintaxis

```
dict(**kwargs)
dict(mapping, **kwargs)
dict(iterable, **kwargs)
```

Crea un nuevo diccionario.

Un diccionario es una estructura de datos que almacena pares clave-valor, donde las claves deben ser únicas e inmutables (por ejemplo, cadenas, números), y los valores pueden ser de cualquier tipo. Los diccionarios permiten un acceso rápido a los valores utilizando claves. Pueden ser modificados dinámicamente añadiendo nuevos pares clave-valor, eliminando los existentes e iterando sobre sus elementos. También se pueden crear diccionarios anidados, lo que permite estructuras de datos más complejas.

Para otros contenedores, consulte las clases integradas `list`, `set` y `tuple`, así como el módulo `collections`.

Ejemplos

- Diccionario vacío `dict()` # `{}`
- Formas de crear un diccionario `dict(one=1, two=2, three=3)` `dict({'one':1, 'two':2, 'three':3})` `dict({'one':1, 'two':2}, three=3)` `dict([('one',1), ('two',2), ('three',3)])` `dict([(('one',1), ('two',2), ('three',3))])` `dict([(('one',1), ('two',2))], three=3)`

1791 clear (builtins.dict) — Elimina todos los elementos del objeto.

ID 1791 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `clear (builtins.dict)`

Explicación española: Elimina todos los elementos del objeto.

Sintaxis

```
clear()
```

Elimina todos los elementos del objeto.

1792 copy (builtins.dict) — Devuelve una copia superficial del objeto.

ID 1792 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `copy (builtins.dict)`
Explicación española: Devuelve una copia superficial del objeto.

Sintaxis

`copy()`

Devuelve una copia superficial del objeto.

1793 fromkeys (builtins.dict) — Crea un nuevo diccionario con claves del iterable y valores establecidos en value.

ID 1793 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `fromkeys (builtins.dict)`
Explicación española: Crea un nuevo diccionario con claves del iterable y valores establecidos en value.

Sintaxis

`dict.fromkeys(iterable [, value])`

Crea un nuevo diccionario con claves del iterable y valores establecidos en value.

`fromkeys()` es un método de clase que devuelve un nuevo diccionario. `value` por defecto es `None`. Todos los valores se refieren a una sola instancia, por lo que generalmente no tiene sentido que `value` sea un objeto mutable como una lista vacía. Para obtener valores distintos, utiliza una comprensión de diccionario en su lugar.

Ejemplo de uso:

```
keys = ['a', 'b', 'c']
```

```
dictionary = dict.fromkeys(keys, 0)
```

1794 get (builtins.dict) — Devuelve el valor de `key` si `key` está en el diccionario, de lo contrario devuelve `default`.

ID 1794 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `get (builtins.dict)`
Explicación española: Devuelve el valor de `key` si `key` está en el diccionario, de lo contrario devuelve `default`.

Sintaxis

`get(key [, default])`

Devuelve el valor de `key` si `key` está en el diccionario, de lo contrario devuelve `default`. Si no se proporciona `default`, se establece de forma predeterminada en `None`, por lo que este método nunca genera un `KeyError`.

1795 items (builtins.dict) — Devuelve una nueva vista de los elementos del diccionario (pares (clave, valor)).

ID 1795 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `items (builtins.dict)`
Explicación española: Devuelve una nueva vista de los elementos del diccionario (pares (clave, valor)).

Sintaxis

`items()`

Devuelve una nueva vista de los elementos del diccionario (pares (clave, valor)). Consulte la documentación de los objetos de vista.

1796 keys (builtins.dict) — Devuelve una nueva vista de las claves del diccionario

ID 1796 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `keys (builtins.dict)`
Explicación española: Devuelve una nueva vista de las claves del diccionario

Sintaxis

`keys()`

Devuelve una nueva vista de las claves del diccionario

1797 pop (builtins.dict) — Si la clave está en el diccionario, se elimina y se devuelve su valor; de lo contrario, se devuelve el...

ID 1797 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `pop (builtins.dict)`

Explicación española: Si la clave está en el diccionario, se elimina y se devuelve su valor; de lo contrario, se devuelve el...

Sintaxis

`pop(key [, default])`

Si la clave está en el diccionario, se elimina y se devuelve su valor; de lo contrario, se devuelve el valor predeterminado. Si no se proporciona un valor predeterminado y la clave no está en el diccionario, se genera un `KeyError`.

1798 popitem (builtins.dict) — Elimina y devuelve un par (clave, valor) del diccionario.

ID 1798 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `popitem (builtins.dict)`

Explicación española: Elimina y devuelve un par (clave, valor) del diccionario.

Sintaxis

`popitem()`

Elimina y devuelve un par (clave, valor) del diccionario. Los pares se devuelven en orden LIFO.

`popitem()` es útil para iterar de manera destructiva sobre un diccionario, como se usa a menudo en algoritmos de conjuntos. Si el diccionario está vacío, llamar a `popitem()` genera un `KeyError`.

1799 setdefault (builtins.dict) — Si la clave está en el diccionario, devuelve su valor.

ID 1799 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `setdefault (builtins.dict)`

Explicación española: Si la clave está en el diccionario, devuelve su valor.

Sintaxis

`setdefault(key [, default])`

Si la clave está en el diccionario, devuelve su valor. Si no, inserta la clave con un valor por defecto y devuelve el valor por defecto. El valor por defecto es `None`.

1800 update (builtins.dict) — Actualiza el diccionario con los pares clave/valor de otro, sobrescribiendo las claves existentes.

ID 1800 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `update (builtins.dict)`

Explicación española: Actualiza el diccionario con los pares clave/valor de otro, sobrescribiendo las claves existentes.

Sintaxis

`update([other])`

Actualiza el diccionario con los pares clave/valor de otro, sobrescribiendo las claves existentes. Devuelve `None`.

`update()` acepta ya sea otro objeto diccionario o un iterable de pares clave/valor (como tuplas u otros iterables de longitud dos). Si se especifican argumentos de palabra clave, el diccionario se actualiza con esos pares clave/valor: `d.update(red=1, blue=2)`.

1801 values (builtins.dict) — Devuelve una nueva vista de los valores del diccionario.

ID 1801 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `values (builtins.dict)`

Explicación española: Devuelve una nueva vista de los valores del diccionario.

Sintaxis

`values()`

Devuelve una nueva vista de los valores del diccionario. Consulte la documentación de los objetos de vista.

Una comparación de igualdad entre una vista `dict.values()` y otra siempre devolverá `False`. Esto también se aplica al comparar `dict.values()` consigo mismo:

1802 enumerate (builtins) — Devuelve un objeto enumerado.

ID 1802 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `enumerate (builtins)`

Explicación española: Devuelve un objeto enumerado.

Sintaxis

`enumerate(i [, start=0])`

Devuelve un objeto enumerado. El iterable `i` debe ser una secuencia, un iterador o algún otro objeto que soporte la iteración. El método `next()` del iterador devuelto por `enumerate()` devuelve una tupla que contiene un contador (a partir de `start`, que por defecto es 0) y los valores obtenidos al iterar sobre el iterable.

1803 filter (builtins) — Construye un iterador a partir de aquellos elementos de `iterable` para los cuales `function` devuelve...

ID 1803 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `filter (builtins)`

Explicación española: Construye un iterador a partir de aquellos elementos de `iterable` para los cuales `function` devuelve...

Sintaxis

`filter(function, iterable)`

Construye un iterador a partir de aquellos elementos de `iterable` para los cuales `function` devuelve verdadero. El `iterable` puede ser una secuencia, un contenedor que admite iteración o un iterador. Si `function` es `None`, se asume la función identidad, es decir, se eliminan todos los elementos de `iterable` que son falsos.

Tenga en cuenta que `filter(function, iterable)` es equivalente a la expresión generadora `(item for item in iterable if function(item))` si `function` no es `None` y `(item for item in iterable if item)` si `function` es `None`.

Ejemplo

```
def is_even(number):  
    return number % 2 == 0
```

```
numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
even_numbers = filter(is_even, numbers)  
print(list(even_numbers)) # Salida: [2, 4, 6, 8, 10]
```

1804 float, clase (builtins)

ID 1804 · título localizado en HelpEs

Sintaxis

`float([x])`

Devuelve un número de punto flotante construido a partir de un número o cadena `x`.

Si el argumento es una cadena, debe contener un número decimal, opcionalmente precedido por un signo, y opcionalmente incrustado en espacios en blanco. El signo opcional puede ser `'+'` o `'-'`; un signo `'+'` no tiene efecto

en el valor producido. El argumento también puede ser una cadena que representa un NaN (not-a-number), o una infinita positiva o negativa. Más precisamente, la entrada debe ajustarse a la siguiente gramática después de eliminar los caracteres de espacio en blanco al principio y al final.

Ejemplos

```
float(5) # 5.0
float(' 7 ') # 7.0
float('nan') # nan
```

1805 frozenset, clase (builtins)

ID 1805 · título localizado en HelpEs

Sintaxis

```
frozenset([iterable])
```

El frozenset es una versión inmutable de un conjunto. Es un tipo de dato de colección que almacena elementos únicos y admite operaciones de conjunto. Dado que es inmutable, una vez creado, no se puede modificar. Esta clase admite muchos de los mismos métodos que el conjunto, pero con la limitación de que los métodos que modifican el conjunto (por ejemplo, add, remove) no están disponibles. El iterable opcional puede ser cualquier objeto iterable (por ejemplo, lista, tupla, conjunto). Si se proporciona, sus elementos se añaden al frozenset.

Ejemplo

```
frozenset((1, 2, 2, 3)) # frozenset({1, 2, 3})
```

1806 copy (builtins.frozens) — Devuelve una copia superficial del frozenset (dado que el frozenset es inmutable, esto devolverá el...

ID 1806 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: copy (builtins.frozens)

Explicación española: Devuelve una copia superficial del frozenset (dado que el frozenset es inmutable, esto devolverá el...

Sintaxis

```
copy()
```

Devuelve una copia superficial del frozenset (dado que el frozenset es inmutable, esto devolverá el mismo objeto frozenset).

Ejemplo

```
fs = frozenset([1, 2, 3])
copy_fs = fs.copy()
print(copy_fs) # Salida: frozenset({1, 2, 3})
```

1807 difference (builtins.frozens) — Devuelve un nuevo frozenset que contiene elementos que están en este frozenset pero no en los otros.

ID 1807 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: difference (builtins.frozens)

Explicación española: Devuelve un nuevo frozenset que contiene elementos que están en este frozenset pero no en los otros.

Sintaxis

```
difference(*others)
frozenset - otro - ...
```

Devuelve un nuevo frozenset que contiene elementos que están en este frozenset pero no en los otros.

Ejemplo

```
fs1 = frozenset([1, 2, 3])
fs2 = frozenset([3, 4])
print(fs1.difference(fs2)) # Salida: frozenset({1, 2})
```

1808 intersection (builtins.frozenset) — Devuelve un nuevo frozenset que contiene los elementos que son comunes entre este frozenset y todos los...

ID 1808 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `intersection (builtins.frozenset)`

Explicación española: Devuelve un nuevo frozenset que contiene los elementos que son comunes entre este frozenset y todos los...

Sintaxis

```
intersection(*others)
frozenset & otro & ...
```

Devuelve un nuevo frozenset que contiene los elementos que son comunes entre este frozenset y todos los demás.

Ejemplo

```
fs1 = frozenset([1, 2, 3])
fs2 = frozenset([2, 3, 4])
print(fs1.intersection(fs2)) # Salida: frozenset({2, 3})
```

1809 isdisjoint (builtins.frozenset) — Devuelve True si el frozenset no tiene elementos en común con otro, de lo contrario devuelve False.

ID 1809 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `isdisjoint (builtins.frozenset)`

Explicación española: Devuelve True si el frozenset no tiene elementos en común con otro, de lo contrario devuelve False.

Sintaxis

```
isdisjoint(other)
```

Devuelve True si el frozenset no tiene elementos en común con otro, de lo contrario devuelve False.

Ejemplo

```
fs1 = frozenset([1, 2, 3])
fs2 = frozenset([4, 5])
print(fs1.isdisjoint(fs2)) # Salida: True
```

1810 issubset (builtins.frozenset) — Verifica si cada elemento en el conjunto está en el otro.

ID 1810 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `issubset (builtins.frozenset)`

Explicación española: Verifica si cada elemento en el conjunto está en el otro.

Sintaxis

```
issubset(other)
frozenset <= other
```

Verifica si cada elemento en el conjunto está en el otro.

`set < other`

Verifica si el conjunto es un subconjunto propio del otro, es decir, `set <= other` y `set != other`.

Ejemplo

```
fs1 = frozenset([1, 2])
fs2 = frozenset([1, 2, 3])
print(fs1.issubset(fs2)) # Salida: True
```

1811 issuperset (builtins.frozenset) — Devuelve True si este frozenset contiene todos los elementos de otro.

ID 1811 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `issuperset` (`builtins.frozens`)

Explicación española: Devuelve True si este frozenset contiene todos los elementos de otro.

Sintaxis

```
issuperset(other)
frozenset >= other
```

Devuelve True si este frozenset contiene todos los elementos de otro.

Ejemplo

```
fs1 = frozenset([1, 2, 3])
fs2 = frozenset([1, 2])
print(fs1.issuperset(fs2)) # Salida: True
```

1812 symmetric_difference (builtins.frozens) — Devuelve un nuevo frozenset que contiene elementos que están en este frozenset o en otro, pero no en...

ID 1812 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `symmetric_difference` (`builtins.frozens`)

Explicación española: Devuelve un nuevo frozenset que contiene elementos que están en este frozenset o en otro, pero no en...

Sintaxis

```
symmetric_difference(other)
frozenset ^ other
```

Devuelve un nuevo frozenset que contiene elementos que están en este frozenset o en otro, pero no en ambos.

Ejemplo

```
fs1 = frozenset([1, 2, 3])
fs2 = frozenset([2, 3, 4])
print(fs1.symmetric_difference(fs2)) # Salida: frozenset({1, 4})
```

1813 union (builtins.frozens) — Devuelve un nuevo frozenset con elementos de este frozenset y de todos los demás.

ID 1813 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `union` (`builtins.frozens`)

Explicación española: Devuelve un nuevo frozenset con elementos de este frozenset y de todos los demás.

Sintaxis

```
union(other)
frozenset | other | ...
```

Devuelve un nuevo frozenset con elementos de este frozenset y de todos los demás.

Ejemplo

```
fs1 = frozenset([1, 2, 3])
fs2 = frozenset([3, 4, 5])
print(fs1.union(fs2)) # Salida: frozenset({1, 2, 3, 4, 5})
```

1814 int, clase (builtins)

ID 1814 · título localizado en HelpEs

Sintaxis

```
int([x [, base=10]])
```

Devuelve un objeto entero construido a partir de un número o cadena x, o devuelve 0 si no se proporcionan argumentos. Si x define `int()`, `int(x)` devuelve `x.int()`. Si x define `index()`, devuelve `x.index()`. Si x define `trunc()`, devuelve `x.trunc()`. Para números de punto flotante, esto trunca hacia cero.

Si x no es un número o si se proporciona una base, entonces x debe ser una cadena, bytes o una instancia de bytearray que represente un literal entero en la base dada. Opcionalmente, el literal puede estar precedido por + o

- (sin espacio entre) y rodeado de espacios en blanco. Un literal de base-n consiste en los dígitos 0 a n-1, con a a z (o A a Z) teniendo valores de 10 a 35. La base predeterminada es 10. Los valores de base permitidos son 0 y 2-36. Los literales de base-2, base-8 y base-16 pueden opcionalmente estar precedidos por 0b/0B, 0o/0O o 0x/0X, similar a los literales enteros en el código. La base 0 significa interpretar la cadena exactamente como un literal entero en el código, permitiendo la detección automática de la base (binaria, octal, decimal o hexadecimal).

Ejemplos

```
print(int('010')) # salida: 10
print(int('010', 8)) # salida: 8
print(int('0o10')) # salida: 8
print(int(2.6)) # salida: 2
```

1815 from_bytes (builtins.int) — Devuelve un entero sin signo representado por el arreglo de bytes dado en el orden de bytes especificado.

ID 1815 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `from_bytes` (`builtins.int`)

Explicación española: Devuelve un entero sin signo representado por el arreglo de bytes dado en el orden de bytes especificado.

Sintaxis

```
int.from_bytes(bytes, byteorder)
```

Devuelve un entero sin signo representado por el arreglo de bytes dado en el orden de bytes especificado.

Parámetros

- `bytes`: Debe ser un objeto similar a bytes (como bytes o bytearray) o un iterable que produzca bytes (como una lista de enteros entre 0 y 255).
- `byteorder`: Determina el orden en el que se interpretan los bytes. Si `byteorder` es "big", el byte más significativo está al principio del arreglo de bytes. Si `byteorder` es "little", el byte más significativo está al final del arreglo de bytes. Para solicitar el orden de bytes nativo del sistema host, use `sys.byteorder` como valor del orden de bytes.

Ejemplo

```
b = bytes([255, 0, 0])
print(int.from_bytes(b, 'big')) # Salida: 16711680
print(int.from_bytes(b, 'little')) # Salida: 255
```

1816 to_bytes (builtins.int) — Devuelve un array de bytes que representa un entero utilizando el orden de bytes especificado.

ID 1816 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `to_bytes` (`builtins.int`)

Explicación española: Devuelve un array de bytes que representa un entero utilizando el orden de bytes especificado.

Sintaxis

```
to_bytes(length, byteorder)
```

Devuelve un array de bytes que representa un entero utilizando el orden de bytes especificado.

El entero se representa utilizando `length` bytes. Se genera un `OverflowError` si el entero no se puede representar con el número dado de bytes.

El argumento `byteorder` determina el orden de bytes utilizado para representar el entero. Si `byteorder` es "big", el byte más significativo está al principio del array de bytes. Si `byteorder` es "little", el byte más significativo está al final del array de bytes. Para solicitar el orden de bytes nativo del sistema host, use `sys.byteorder` como valor del orden de bytes.

Ejemplos

```
(1024).to_bytes(2, 'big')
b'\x04\x00'
```

```
(1024).to_bytes(2, 'little')
b'\x00\x04'
```

```
(1024).to_bytes(10, 'big')
b'\x00\x00\x00\x00\x00\x00\x00\x00\x04\x00'
```

1817 list, clase (builtins)

ID 1817 · título localizado en HelpEs

Sintaxis

```
list([iterable])
```

La clase lista representa una secuencia mutable de elementos. Es una de las estructuras de datos fundamentales en Python, que permite el almacenamiento y la manipulación de colecciones de elementos. Las listas son ordenadas, indexadas y pueden contener elementos de diferentes tipos.

1818 append (builtins.list) — Agrega el elemento x al final de la lista.

ID 1818 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `append (builtins.list)`
Explicación española: Agrega el elemento x al final de la lista.

Sintaxis

```
append(x)
```

Agrega el elemento x al final de la lista. El x puede ser cualquier objeto y se agrega a la lista en su lugar, modificando la lista original.

Ejemplo

```
my_list = [1, 2, 3]
my_list.append('ABC')
print(my_list) # Salida: [1, 2, 3, 'ABC']
```

1819 clear (builtins.list) — Elimina todos los elementos de la lista, dejándola vacía.

ID 1819 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `clear (builtins.list)`
Explicación española: Elimina todos los elementos de la lista, dejándola vacía.

Sintaxis

```
clear()
```

Elimina todos los elementos de la lista, dejándola vacía. Esta es una operación en el lugar (mutación), por lo que la lista original se modifica.

Ejemplo

```
my_list = [1, 2, 3, 4]
my_list.clear()
print(my_list) # Salida: []
```

1820 copy (builtins.list) — Devuelve una copia superficial de la lista.

ID 1820 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `copy (builtins.list)`
Explicación española: Devuelve una copia superficial de la lista.

Sintaxis

```
copy()
```

Devuelve una copia superficial de la lista. La lista original permanece sin cambios. Las modificaciones a la lista copiada no afectan a la lista original, pero los objetos anidados (como listas o diccionarios) dentro de la lista siguen siendo referencias a los mismos objetos en ambas listas (copia superficial).

Ejemplo

```
my_list = [[1, 2], 3]
copied_list = my_list.copy()
print(copied_list) # salida: [[1, 2], 3]
# Modificar la lista anidada en la lista copiada
copied_list[0][1] = 'A'
print("Lista original:", my_list) # Salida: [[1, 'A'], 3]
print("Lista copiada:", copied_list) # Salida: [[1, 'A'], 3]
```

1821 count (builtins.list) — Devuelve el número de ocurrencias del elemento x en la lista.

ID 1821 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: count (builtins.list)

Explicación española: Devuelve el número de ocurrencias del elemento x en la lista.

Sintaxis

count(x)

Devuelve el número de ocurrencias del elemento x en la lista. El elemento x puede ser de cualquier tipo, incluidas otras listas. Cuando x es una lista u otro objeto complejo, el método verifica si hay una coincidencia exacta, lo que significa que solo cuenta las instancias donde x es idéntico a un objeto específico en la lista (no solo que contenga los mismos elementos).

Ejemplo

```
my_list = [1, 2, 3, 2, 2, 4]
count_of_twos = my_list.count(2)
print(count_of_twos) # Salida: 3
```

1822 extend (builtins.list) — Extiende la lista al agregar todos los elementos del iterable dado.

ID 1822 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: extend (builtins.list)

Explicación española: Extiende la lista al agregar todos los elementos del iterable dado.

Sintaxis

extend(iterable)

Extiende la lista al agregar todos los elementos del iterable dado. El iterable puede ser cualquier objeto que produzca elementos, como otra lista, una tupla o un conjunto. Esta es una operación en el lugar (mutación), lo que significa que la lista original se modifica.

Ejemplo

```
my_list = [1, 2, 3]
my_list.extend([4, 5])
print(my_list) # Salida: [1, 2, 3, 4, 5]
```

1823 index (builtins.list) — Devuelve el índice de la primera ocurrencia del elemento x en la lista.

ID 1823 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: index (builtins.list)

Explicación española: Devuelve el índice de la primera ocurrencia del elemento x en la lista.

Sintaxis

index(x [, start [, end]])

Devuelve el índice de la primera ocurrencia del elemento x en la lista. Si no se encuentra x, se genera un ValueError. Opcionalmente, puede especificar el rango de búsqueda con los parámetros start y end, que definen el segmento de la lista a buscar.

Ejemplo

```
my_list = [1, 2, 3, 4, 2]
index = my_list.index(2)
print(index) # Salida: 1
# Usando los parámetros opcionales start y end
index = my_list.index(2, 2)
print(index) # Salida: 4
```

1824 insert (builtins.list) — A el elemento x en el índice especificado i.

ID 1824 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: insert (builtins.list)

Explicación española: A el elemento x en el índice especificado i.

Sintaxis

insert(i, x)

Inserta el elemento x en el índice especificado i. El índice puede ser cualquier entero, donde i = 0 inserta al principio de la lista, y se pueden usar índices negativos para insertar en relación con el final de la lista. Esta es una operación en el lugar (mutación) que modifica la lista original.

Ejemplo

```
my_list = [1, 2, 4]
my_list.insert(2, 3)
print(my_list) # Salida: [1, 2, 3, 4]
```

1825 pop (builtins.list) — Elimina y devuelve el elemento en el índice especificado i de la lista.

ID 1825 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: pop (builtins.list)

Explicación española: Elimina y devuelve el elemento en el índice especificado i de la lista.

Sintaxis

pop([i])

Elimina y devuelve el elemento en el índice especificado i de la lista. Si no se proporciona i, pop() elimina y devuelve el último elemento de la lista. La lista se modifica en su lugar. Si el índice especificado está fuera de rango, se genera un IndexError.

Ejemplo

```
my_list = [1, 2, 3, 4]
# Eliminar el último elemento
last_element = my_list.pop()
print(last_element) # Salida: 4
print(my_list) # Salida: [1, 2, 3]
# Eliminar el elemento en el índice 1
second_element = my_list.pop(1)
print(second_element) # Salida: 2
print(my_list) # Salida: [1, 3]
```

1826 remove (builtins.list) — Elimina la primera ocurrencia del elemento x de la lista.

ID 1826 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: remove (builtins.list)

Explicación española: Elimina la primera ocurrencia del elemento x de la lista.

Sintaxis

remove(x)

Elimina la primera ocurrencia del elemento x de la lista. Si no se encuentra x, se genera un ValueError. Esta operación modifica la lista en su lugar y no devuelve nada.

Ejemplo

```
my_list = [1, 2, 3, 2, 4]
# Eliminar la primera ocurrencia de 2
my_list.remove(2)
print(my_list) # Salida: [1, 3, 2, 4]
```

1827 reverse (builtins.list) — Invierte los elementos de la lista en su lugar, lo que significa que la lista se modifica directamente y...

ID 1827 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `reverse (builtins.list)`
Explicación española: Invierte los elementos de la lista en su lugar, lo que significa que la lista se modifica directamente y...

Sintaxis

`reverse()`

Invierte los elementos de la lista en su lugar, lo que significa que la lista se modifica directamente y el orden de sus elementos se invierte. Esta operación no devuelve una nueva lista, sino que modifica la lista original.

Ejemplo

```
my_list = [1, 2, 3, 4]
my_list.reverse()
print(my_list) # Salida: [4, 3, 2, 1]
```

1828 sort (builtins.list) — Ordena la lista en orden ascendente por defecto.

ID 1828 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `sort (builtins.list)`
Explicación española: Ordena la lista en orden ascendente por defecto.

Sintaxis

```
sort()
sort(reverse=False)
sort(key=Func [, reverse=False])
```

Ordena la lista en orden ascendente por defecto. La ordenación se realiza en su lugar, lo que significa que la lista original se modifica. El parámetro `reverse` permite ordenar en orden descendente si se establece en `True`. Se puede proporcionar el parámetro `key` para especificar una función personalizada que se utilizará para las comparaciones de ordenación.

Ejemplo

```
# Ordenando una lista de números
my_list = [3, 1, 4, 2]
my_list.sort()
print(my_list) # Salida: [1, 2, 3, 4]
# Ordenando en orden inverso
my_list.sort(reverse=True)
print(my_list) # Salida: [4, 3, 2, 1]
```

1829 map (builtins) — (función, iterable, ...)

ID 1829 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `map (builtins)`
Explicación española: (función, iterable, ...)

`map(función, iterable, ...)`

Devuelve un iterador que aplica la función a cada elemento del iterable, produciendo los resultados. Si se pasan argumentos iterables adicionales, la función debe tomar esa cantidad de argumentos y se aplica a los elementos de todos los iterables en paralelo. Con múltiples iterables, el iterador se detiene cuando el iterable más corto se agota.

1830 memoryview, clase (builtins)

ID 1830 · título localizado en HelpEs

Sintaxis

`memoryview(obj)`

Crea un `memoryview` que referencia el objeto `obj`. El `obj` debe soportar el protocolo de buffer. Los objetos integrados que soportan el protocolo de buffer incluyen `bytes` y `bytearray`.

Un `memoryview` tiene la noción de un elemento, que es la unidad de memoria atómica manejada por el objeto original `obj`. Para muchos tipos simples como `bytes` y `bytearray`, un elemento es un solo byte, pero otros tipos como `array.array` pueden tener elementos más grandes.

Si tienes un `bytearray` y creas un `memoryview` `mv` que visualiza todo el `bytearray`, entonces `len(mv)` devolverá la longitud del `bytearray`. Si el `memoryview` solo cubre una parte del `bytearray`, `len(mv)` devolverá la longitud de esa parte.

Ejemplo

```
# Crear un bytearray a partir de una cadena
data = bytearray(b"Hello World")
# Crear un memoryview sobre el bytearray
view = memoryview(data)
# Leer y mostrar parte
# del bytearray usando memoryview
print(bytes(view[6:11])) # b'World'
# Modificar parte del bytearray
view[6:11] = b"Prime"
# Mostrar el bytearray modificado
# Esto mostrará "Hello Prime"
print(data)
```

1894 NotImplemented — Es una constante especial en Python utilizada para indicar que una operación o método particular no está...

ID 1894 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `NotImplemented`

Explicación española: Es una constante especial en Python utilizada para indicar que una operación o método particular no está...

`NotImplemented` es una constante especial en Python utilizada para indicar que una operación o método particular no está implementado para el operando o contexto dado.

Generalmente es devuelta por métodos especiales (como `__eq__`, `__lt__`, etc.) cuando el método no admite la operación con los argumentos proporcionados. Devolver `NotImplemented` permite a Python intentar métodos alternativos o volver al comportamiento predeterminado.

Es diferente de `NotImplementedError`, que señala que un método o función aún no está implementado.

1831 object, clase (builtins)

ID 1831 · título localizado en HelpEs

Sintaxis

`object()`

Devuelve un nuevo objeto sin características. `object` es la base de todas las clases. Tiene los métodos que son comunes a todas las instancias de las clases de Python. Esta función no acepta ningún argumento.

1832 property, clase (builtins)

ID 1832 · título localizado en HelpEs

Sintaxis

`property(fget=None, fset=None, fdel=None, doc=None)`

Devuelve un atributo de propiedad. `fget` es una función para obtener el valor de un atributo, `fset` es una función para establecer el valor de un atributo, y `fdel` es una función para eliminar el valor de un atributo. En esta implementación de MicroPython, la cadena de documentación para la propiedad no es compatible, incluso si se

proporciona a través del argumento doc o utilizando decoradores.

Un uso típico es definir un atributo gestionado x:

```
class C:
    def init(self):
        self._x = None

    def getx(self):
        return self._x

    def setx(self, value):
        self._x = value

    def delx(self):
        del self._x

x = property(getx, setx, delx)
```

Si c es una instancia de C, c.x invocará el getter, c.x = value invocará el setter, y del c.x invocará el deleter.

También puedes crear propiedades usando decoradores:

```
class Parrot:
    def __init__(self):
        self._voltage = 100000

    @property
    def voltage(self):
        return self._voltage

    @voltage.setter
    def voltage(self, value):
        self._voltage = value

    @voltage.deleter
    def voltage(self):
        del self._voltage
```

El decorador @property convierte el método voltage() en un getter para un atributo de solo lectura y permite que el setter y el deleter se agreguen más tarde.

1833 getter (builtins.property) — El método getter asigna o modifica la función getter para una propiedad.

ID 1833 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `getter (builtins.property)`

Explicación española: El método getter asigna o modifica la función getter para una propiedad.

Sintaxis

`getter(fget)`

El método getter asigna o modifica la función getter para una propiedad. Toma un argumento, fget, que es la función responsable de recuperar el valor de la propiedad. Este método devuelve un nuevo objeto de propiedad con la función getter actualizada.

Ejemplo

```
class MyClass:
    def __init__(self):
        self._value = 42
```

```
@property
def value(self):
    return self._value

@value.getter
def value(self):
    """Getter para la propiedad 'value'."""
    return self._value

obj = MyClass()
print(obj.value) # Salida: 42
```

1834 setter (builtins.property) — El método setter asigna o modifica la función setter para una propiedad.

Referencia traducidaIdentificador técnico: `setter` (`builtins.property`)Explicación española: El método `setter` asigna o modifica la función `setter` para una propiedad.**Sintaxis**`setter(fset)`

El método `setter` asigna o modifica la función `setter` para una propiedad. Toma un argumento, `fset`, que es la función responsable de establecer el valor de la propiedad. Este método devuelve un nuevo objeto de propiedad con la función `setter` actualizada.

```
class MyClass:
    def __init__(self):
        self._value = 42

    @property
    def value(self):
        return self._value

    @value.setter
    def value(self, new_value):
        self._value = new_value

# Creando una instancia de MyClass
obj = MyClass()
print(obj.value) # Salida: 42

# Estableciendo un nuevo valor
obj.value = 100
print(obj.value) # Salida: 100
```

1835 deleter (`builtins.property`) — El método `deleter` asigna o modifica la función de eliminación para una propiedad.

Referencia traducidaIdentificador técnico: `deleter` (`builtins.property`)Explicación española: El método `deleter` asigna o modifica la función de eliminación para una propiedad.**Sintaxis**`deleter(fdel)`

El método `deleter` asigna o modifica la función de eliminación para una propiedad. Toma un argumento, `fdel`, que es la función responsable de eliminar el valor de la propiedad. Este método devuelve un nuevo objeto de propiedad con la función de eliminación actualizada.

1836 range, clase (`builtins`)

Sintaxis

```
range(stop)
range(start, stop [, step])
```

La clase `range` representa una secuencia inmutable de números y se utiliza comúnmente para generar una secuencia de enteros, especialmente en bucles. Esta clase es eficiente en términos de uso de memoria, ya que genera números bajo demanda en lugar de almacenarlos todos en memoria. El parámetro `start` especifica el comienzo de la secuencia, `stop` indica el final de la secuencia y `step` determina el incremento entre cada número en la secuencia. El parámetro `stop` es obligatorio, mientras que `start` por defecto es 0 y `step` por defecto es 1 si no se especifica.

Ejemplos

```
range(5) # genera [0, 1, 2, 3, 4]
range(2,6) # genera [2, 3, 4, 5]
range(0,20,5) # genera [0, 5, 10, 15]
```

1837 reversed (builtins) — La función reversed devuelve un iterador que le permite acceder a los elementos de una secuencia en...

ID 1837 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: reversed (builtins)

Explicación española: La función reversed devuelve un iterador que le permite acceder a los elementos de una secuencia en...

Sintaxis

reversed(seq)

La función reversed devuelve un iterador que le permite acceder a los elementos de una secuencia en orden inverso, sin crear una nueva lista. Es compatible con secuencias como listas, tuplas y cadenas, pero no admite diccionarios ni OrderedDict, que generalmente no están diseñados para mantener el orden o ser invertidos de manera eficiente en MicroPython.

Para usar reversed, simplemente pase la secuencia seq que desea invertir. Por ejemplo, invertir una lista de números implicaría llamar a reversed() en la lista e iterar sobre el resultado para acceder a los elementos de último a primero. Este enfoque no duplica la lista, sino que proporciona directamente una capacidad de iteración hacia atrás.

Ejemplo

```
# Invertiendo una lista
numbers = [1, 2, 3, 4, 5]
for number in reversed(numbers):
    print(number) # Salida: 5, 4, 3, 2, 1
```

1838 set, clase (builtins)

ID 1838 · título localizado en HelpEs

Sintaxis

set([i])

La clase set representa una colección desordenada de elementos únicos. Se utiliza para almacenar múltiples elementos en una sola variable, asegurando que no se incluyan elementos duplicados. Si se proporciona un iterable i, el conjunto se crea a partir de sus elementos, eliminando cualquier duplicado. Los conjuntos son mutables, lo que significa que puede agregar o eliminar elementos después de la creación, pero no admiten indexación, segmentación ni otro comportamiento similar a secuencias.

El set es una clase incorporada. Consulte set y tipos de Set — set, frozenset para obtener documentación sobre esta clase.

1839 add (builtins.set) — Agrega el elemento especificado `e` al conjunto.

ID 1839 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: add (builtins.set)

Explicación española: Agrega el elemento especificado `e` al conjunto.

Sintaxis

add(e)

Agrega el elemento especificado e al conjunto. Si el elemento ya está presente en el conjunto, no sucede nada (no se permiten duplicados en un conjunto). Esta operación se realiza en su lugar y modifica el conjunto original.

Ejemplo

```
s = {1, 2, 3}
s.add(4) # Agregando un nuevo elemento
print(s) # Salida: {1, 2, 3, 4}
s.add(2) # Intentando agregar un elemento existente
print(s) # Salida: {1, 2, 3, 4}
```

1840 clear (builtins.set) — Elimina todos los elementos del conjunto, dejándolo vacío.

Referencia traducidaIdentificador técnico: `clear (builtins.set)`

Explicación española: Elimina todos los elementos del conjunto, dejándolo vacío.

Sintaxis`clear()`

Elimina todos los elementos del conjunto, dejándolo vacío. Esta operación se realiza en su lugar y no devuelve un valor.

Ejemplo

```
s = {1, 2, 3, 4}
s.clear() # Limpiando el conjunto
print(s) # Salida: set()
```

1841 copy (builtins.set) — Devuelve una copia superficial del conjunto.

Referencia traducidaIdentificador técnico: `copy (builtins.set)`

Explicación española: Devuelve una copia superficial del conjunto.

Sintaxis`copy()`

Devuelve una copia superficial del conjunto. La copia contiene todos los elementos del conjunto original, pero es un objeto distinto. Esta operación no modifica el conjunto original.

Ejemplo

```
s = {1, 2, 3, 4}
s_copy = s.copy() # Creando una copia
print(s_copy) # Salida: {1, 2, 3, 4}
print(s is s_copy) # Salida: False
```

1843 difference (builtins.set) — Devuelve un nuevo conjunto que contiene elementos que están presentes en el conjunto original pero no en...

Referencia traducidaIdentificador técnico: `difference (builtins.set)`

Explicación española: Devuelve un nuevo conjunto que contiene elementos que están presentes en el conjunto original pero no en...

Sintaxis`difference(*others)`

Devuelve un nuevo conjunto que contiene elementos que están presentes en el conjunto original pero no en ninguno de los otros conjuntos (*others). Esta operación no modifica el conjunto original, se devuelve un nuevo conjunto.

Ejemplo

```
s1 = {1, 2, 3, 4}
s2 = {3, 4, 5, 6}
print(s1.difference(s2)) # Salida: {1, 2}
```

1844 difference_update (builtins.set) — Elimina los elementos del conjunto original que también están presentes en cualquiera de los otros...

Referencia traducida

Identificador técnico: `difference_update` (`builtins.set`)

Explicación española: Elimina los elementos del conjunto original que también están presentes en cualquiera de los otros...

Sintaxis

```
difference_update(*others)
set -= other | ...
```

Elimina los elementos del conjunto original que también están presentes en cualquiera de los otros conjuntos (*others). Esta operación se realiza en su lugar y modifica el conjunto original. A diferencia de `difference()`, este método no devuelve un nuevo conjunto, sino que actualiza directamente el conjunto original.

Ejemplo

```
s1 = {1, 2, 3, 4}
s2 = {3, 4, 5, 6}
s1.difference_update(s2)
print(s1) # Salida: {1, 2}
```

1842 discard (builtins.set) — Elimina el elemento especificado `e` del conjunto si está presente.

ID 1842 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `discard` (`builtins.set`)

Explicación española: Elimina el elemento especificado `e` del conjunto si está presente.

Sintaxis

```
discard(e)
```

Elimina el elemento especificado `e` del conjunto si está presente. Si el elemento no está presente, no se genera ningún error y no sucede nada. Esta operación se realiza en su lugar y modifica el conjunto original.

Ejemplo

```
s = {1, 2, 3, 4}
s.discard(3)
print(s) # Salida: {1, 2, 4}
```

1845 intersection (builtins.set) — Devuelve un nuevo conjunto que contiene los elementos que son comunes tanto al conjunto original como al...

ID 1845 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `intersection` (`builtins.set`)

Explicación española: Devuelve un nuevo conjunto que contiene los elementos que son comunes tanto al conjunto original como al...

Sintaxis

```
intersection(other)
```

Devuelve un nuevo conjunto que contiene los elementos que son comunes tanto al conjunto original como al conjunto `other`. Esta operación no modifica el conjunto original; en su lugar, se devuelve un nuevo conjunto.

Ejemplo

```
s1 = {1, 2, 3, 4}
s2 = {3, 4, 5, 6}
print(s1.intersection(s2)) # Salida: {3, 4}
```

1846 intersection_update (builtins.set) — Actualiza el conjunto original para mantener solo los elementos que son comunes tanto al conjunto...

ID 1846 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `intersection_update` (`builtins.set`)

Explicación española: Actualiza el conjunto original para mantener solo los elementos que son comunes tanto al conjunto...

Sintaxis

```
intersection_update(other)
set &= other & ...
```

Actualiza el conjunto original para mantener solo los elementos que son comunes tanto al conjunto original como al conjunto `other`. Esta operación modifica el conjunto original en su lugar y no devuelve un nuevo conjunto.

Ejemplos

```
s1 = {1, 2, 3, 4}
s2 = {3, 4, 5, 6}
s1.intersection_update(s2)
print(s1) # Salida: {3, 4}
```

1847 `isdisjoint` (`builtins.set`) — Devuelve `True` si el conjunto original no tiene elementos en común con el conjunto `other`.

ID 1847 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `isdisjoint` (`builtins.set`)

Explicación española: Devuelve `True` si el conjunto original no tiene elementos en común con el conjunto `other`.

Sintaxis

```
isdisjoint(other)
```

Devuelve `True` si el conjunto original no tiene elementos en común con el conjunto `other`. De lo contrario, devuelve `False`. Esta operación no modifica ninguno de los conjuntos.

Ejemplo

```
s1 = {1, 2, 3}
s2 = {4, 5, 6}
print(s1.isdisjoint(s2)) # Salida: True
```

1848 `issubset` (`builtins.set`) — Devuelve `True` si todos los elementos del conjunto original están contenidos en el conjunto `other`.

ID 1848 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `issubset` (`builtins.set`)

Explicación española: Devuelve `True` si todos los elementos del conjunto original están contenidos en el conjunto `other`.

Sintaxis

```
issubset(other)
```

Devuelve `True` si todos los elementos del conjunto original están contenidos en el conjunto `other`. De lo contrario, devuelve `False`.

Ejemplo

```
s1 = {1, 2}
s2 = {1, 2, 3, 4}
print(s1.issubset(s2)) # Salida: True
```

1849 `issuperset` (`builtins.set`) — Devuelve `True` si el conjunto original contiene todos los elementos del conjunto `other`.

ID 1849 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `issuperset` (`builtins.set`)

Explicación española: Devuelve `True` si el conjunto original contiene todos los elementos del conjunto `other`.

Sintaxis

```
issuperset(other)
```

Devuelve True si el conjunto original contiene todos los elementos del conjunto other. De lo contrario, devuelve False. Esta operación no modifica ninguno de los conjuntos.

Ejemplo

```
s1 = {1, 2, 3, 4}
s2 = {1, 2}
print(s1.issuperset(s2)) # Salida: True
```

1850 pop (builtins.set) — Elimina y devuelve un elemento arbitrario del conjunto.

ID 1850 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: pop (builtins.set)

Explicación española: Elimina y devuelve un elemento arbitrario del conjunto.

Sintaxis

```
pop()
```

Elimina y devuelve un elemento arbitrario del conjunto. Esta operación modifica el conjunto original en su lugar. Si el conjunto está vacío, genera un KeyError.

Ejemplo

```
s = {1, 2, 3, 4}
removed_element = s.pop()
print(removed_element) # Salida: (Un elemento arbitrario, por ejemplo, 1)
print(s) # Salida: {2, 3, 4}
```

1851 remove (builtins.set) — Elimina el elemento especificado e del conjunto.

ID 1851 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: remove (builtins.set)

Explicación española: Elimina el elemento especificado e del conjunto.

Sintaxis

```
remove(e)
```

Elimina el elemento especificado e del conjunto. Si el elemento no está presente en el conjunto, se genera un KeyError. Esta operación modifica el conjunto original en su lugar.

Ejemplo

```
s = {1, 2, 3, 4}
s.remove(3)
print(s) # Salida: {1, 2, 4}
```

1852 symmetric_difference (builtins.set) — Devuelve un nuevo conjunto que contiene elementos que están en el conjunto original o en el conjunto...

ID 1852 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: symmetric_difference (builtins.set)

Explicación española: Devuelve un nuevo conjunto que contiene elementos que están en el conjunto original o en el conjunto...

Sintaxis

```
symmetric_difference(other)
```

Devuelve un nuevo conjunto que contiene elementos que están en el conjunto original o en el conjunto other, pero no en ambos. Esta operación no modifica el conjunto original; en su lugar, devuelve un nuevo conjunto con la diferencia simétrica.

Ejemplo

```
s1 = {1, 2, 3, 4}
s2 = {3, 4, 5, 6}
print(s1.symmetric_difference(s2)) # Salida: {1, 2, 5, 6}
```

1853 symmetric_difference_update (builtins.set) — Actualiza el conjunto original para contener la diferencia simétrica entre el conjunto original y el...

ID 1853 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `symmetric_difference_update` (`builtins.set`)

Explicación española: Actualiza el conjunto original para contener la diferencia simétrica entre el conjunto original y el...

Sintaxis

```
symmetric_difference_update(other)
set ^= other
```

Actualiza el conjunto original para contener la diferencia simétrica entre el conjunto original y el conjunto `other`. La diferencia simétrica incluye elementos que están en el conjunto original o en el otro conjunto, pero no en ambos. Esta operación modifica el conjunto original en su lugar y no devuelve un nuevo conjunto.

Ejemplo

```
s1 = {1, 2, 3, 4}
s2 = {3, 4, 5, 6}
s1.symmetric_difference_update(s2)
print(s1) # Salida: {6, 1, 2, 5}
```

1854 union (builtins.set) — Devuelve un nuevo conjunto que contiene todos los elementos únicos del conjunto original y del conjunto...

ID 1854 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `union` (`builtins.set`)

Explicación española: Devuelve un nuevo conjunto que contiene todos los elementos únicos del conjunto original y del conjunto...

Sintaxis

```
union(other)
```

Devuelve un nuevo conjunto que contiene todos los elementos únicos del conjunto original y del conjunto `other`. El método `union()` no modifica el conjunto original, sino que devuelve un nuevo conjunto con los elementos combinados.

Ejemplo

```
s1 = {1, 2, 3}
s2 = {3, 4, 5}
print(s1.union(s2)) # Salida: {1, 2, 3, 4, 5}
```

1855 update (builtins.set) — Actualiza el conjunto original añadiendo todos los elementos únicos del conjunto `other`.

ID 1855 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `update` (`builtins.set`)

Explicación española: Actualiza el conjunto original añadiendo todos los elementos únicos del conjunto `other`.

Sintaxis

```
update(other)
```

Actualiza el conjunto original añadiendo todos los elementos únicos del conjunto `other`. Esta operación modifica el conjunto original en su lugar y no devuelve un nuevo conjunto.

Ejemplo

```
s1 = {1, 2, 3}
s2 = {3, 4, 5}
s1.update(s2)
print(s1) # Salida: {1, 2, 3, 4, 5}
```

1856 slice, class (builtins)

ID 1856 · título localizado en HelpEs

Sintaxis

```
slice(stop)
slice(start, stop [, step])
```

Devuelve un objeto de corte que representa el conjunto de índices especificados por `range(start, stop, step)`. Los argumentos `start` y `step` son `None` por defecto. Los objetos de corte tienen atributos de datos de solo lectura `start`, `stop` y `step` que simplemente devuelven los valores de los argumentos (o sus valores predeterminados). No tienen otra funcionalidad explícita; sin embargo, son utilizados por Numerical Python y otras extensiones de terceros. Los objetos de corte también se generan cuando se utiliza la sintaxis de indexación extendida. Por ejemplo: `a[start:stop:step]` o `a[start:stop, i]`.

1857 str, class (builtins)

ID 1857 · título localizado en HelpEs

Sintaxis

```
str([object=' ' [, encoding='utf-8' [, errors='strict']]])
```

La clase `str` representa cadenas, que son secuencias de caracteres. Es un tipo de dato inmutable, lo que significa que una vez que se crea una cadena, no se puede cambiar; cualquier operación que modifique el texto siempre devuelve un nuevo objeto `str`.

Las cadenas se pueden crear encerrando caracteres en comillas simples (`'`) o comillas dobles (`"`). Ambos tipos de comillas se tratan por igual, pero las comillas deben ser consistentes en ambos extremos de la cadena.

La clase `str` también se puede utilizar para convertir otros objetos a su representación en cadena. Cuando un objeto `object` se pasa a `str()`, se llama al método `__str__()` de ese objeto para generar su forma de cadena. Si no se proporciona ningún argumento, se devuelve una cadena vacía.

Aunque los parámetros `encoding` y `errors` están presentes, se ignoran. Al convertir de bytes o bytearray, la codificación predeterminada es `'utf-8'` y cualquier error de codificación se maneja estrictamente (`'strict'`), lo que significa que los datos no válidos generarán un error.

Ejemplos

```
# Creando una cadena
s1 = 'hello'
s2 = "world"
```

```
# Convirtiendo objetos a cadenas
str(42) # '42'
str(3.14) # '3.14'
str([1,2,3]) # '[1, 2, 3]'
str({'key': 'value'}) # '{"key": "value"}'
str(bytes([104,101,108,108,111]),'utf-8') # 'hello'
```

1858 encode (builtins.str) — Devuelve una versión codificada de la cadena como un objeto de bytes.

ID 1858 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `encode (builtins.str)`

Explicación española: Devuelve una versión codificada de la cadena como un objeto de bytes.

Sintaxis

```
encode([encoding='utf-8' [, errors='strict']])
```

Devuelve una versión codificada de la cadena como un objeto de bytes. Aunque los parámetros de codificación y errores están presentes, se ignoran. La cadena siempre se codifica utilizando la codificación `'utf-8'`, y cualquier error de codificación se maneja utilizando la política de manejo de errores `'strict'`, lo que significa que los

caracteres no válidos generarán un `UnicodeError`.

1859 find (builtins.str) — Devuelve el índice más bajo en la cadena donde se encuentra la subcadena ``sub``, de modo que ``sub`` esté...

ID 1859 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `find (builtins.str)`

Explicación española: Devuelve el índice más bajo en la cadena donde se encuentra la subcadena ``sub``, de modo que ``sub`` esté...

Sintaxis

```
find(sub [, start [, end]])
```

Devuelve el índice más bajo en la cadena donde se encuentra la subcadena `sub`, de modo que `sub` esté contenido dentro de la porción `s[start:end]`. Los argumentos opcionales `start` y `end` se interpretan como en la notación de porciones. Devuelve `-1` si no se encuentra `sub`.

1860 rfind (builtins.str) — Devuelve el índice más alto en la cadena donde se encuentra la subcadena ``sub``, de modo que ``sub`` esté...

ID 1860 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `rfind (builtins.str)`

Explicación española: Devuelve el índice más alto en la cadena donde se encuentra la subcadena ``sub``, de modo que ``sub`` esté...

Sintaxis

```
rfind(sub [, start [, end]])
```

Devuelve el índice más alto en la cadena donde se encuentra la subcadena `sub`, de modo que `sub` esté contenido dentro de `s[start:end]`. Los argumentos opcionales `start` y `end` se interpretan como en la notación de rebanado. Devuelve `-1` en caso de fallo.

1861 index (builtins.str) — Funciona como `find()`, pero genera un `ValueError` cuando no se encuentra la subcadena ``sub``.

ID 1861 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `index (builtins.str)`

Explicación española: Funciona como `find()`, pero genera un `ValueError` cuando no se encuentra la subcadena ``sub``.

Sintaxis

```
index(sub [, start [, end]])
```

Funciona como `find()`, pero genera un `ValueError` cuando no se encuentra la subcadena `sub`.

1862 rindex (builtins.str) — Funciona como `rfind()`, pero genera un `ValueError` cuando no se encuentra la subcadena ``sub``.

ID 1862 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `rindex (builtins.str)`

Explicación española: Funciona como `rfind()`, pero genera un `ValueError` cuando no se encuentra la subcadena ``sub``.

Sintaxis

```
rindex(sub [, start [, end]])
```

Funciona como `rfind()`, pero genera un `ValueError` cuando no se encuentra la subcadena `sub`.

1863 join (builtins.str) — Devuelve una cadena que es la concatenación de las cadenas en el iterable ``iter``.

ID 1863 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `join` (`builtins.str`)

Explicación española: Devuelve una cadena que es la concatenación de las cadenas en el iterable `iter`.

Sintaxis

`join(iter)`

Devuelve una cadena que es la concatenación de las cadenas en el iterable `iter`. Se generará un `TypeError` si hay valores en el iterable que no son objetos de cadena. El separador entre los elementos es el contenido del objeto de cadena que proporciona este método.

1864 `split` (`builtins.str`) — Divide la cadena en una lista de subcadenas, utilizando `sep` como la cadena delimitadora.

ID 1864 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `split` (`builtins.str`)

Explicación española: Divide la cadena en una lista de subcadenas, utilizando `sep` como la cadena delimitadora.

Sintaxis

`split([sep=None [, maxsplit=-1]])`

Divide la cadena en una lista de subcadenas, utilizando `sep` como la cadena delimitadora. Si se proporciona `maxsplit` y es no negativo, se realizan como máximo `maxsplit` divisiones, resultando en una lista con como máximo `maxsplit+1` elementos. Si `maxsplit` no se especifica o es `-1`, no hay límite en el número de divisiones (se realizan todas las divisiones posibles).

Si se especifica `sep`, los delimitadores consecutivos no se agrupan y se consideran como delimitadores de subcadenas vacías (por ejemplo, `'1,,2'.split(',')` devuelve `['1', '', '2']`). El argumento `sep` puede ser cualquier cadena, incluidas las cadenas de múltiples caracteres (por ejemplo, `'1{2}3'.split('{}')` devuelve `['1', '2', '3']`). Dividir una cadena vacía con un separador especificado devuelve `['']`.

Si `sep` no se especifica o es `None`, se aplica un algoritmo de división diferente: las secuencias de espacios en blanco consecutivos se consideran como un solo separador, y el resultado no contendrá cadenas vacías al principio o al final si la cadena tiene espacios en blanco al principio o al final. Por lo tanto, dividir una cadena vacía o una cadena que consiste únicamente en espacios en blanco sin un separador especificado devuelve `[]`.

1865 `splitlines` (`builtins.str`) — Devuelve una lista de las líneas en la cadena, rompiendo en los límites de línea.

ID 1865 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `splitlines` (`builtins.str`)

Explicación española: Devuelve una lista de las líneas en la cadena, rompiendo en los límites de línea.

Sintaxis

`splitlines([keepends=False])`

Devuelve una lista de las líneas en la cadena, rompiendo en los límites de línea. Este método utiliza el enfoque de nuevas líneas universales para dividir líneas. Los saltos de línea no se incluyen en la lista resultante a menos que se especifique `keepends` y se establezca en `true`.

Ejemplo

```
'Hello\nWorld'.splitlines(True) # ['Hello\n', 'World'].
```

1866 `rsplit` (`builtins.str`) — Divide la cadena en una lista de subcadenas, utilizando `sep` como la cadena delimitadora, comenzando...

ID 1866 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `rsplit` (`builtins.str`)

Explicación española: Divide la cadena en una lista de subcadenas, utilizando `sep` como la cadena delimitadora, comenzando...

Sintaxis

`rsplit([sep=None [, maxsplit=-1]])`

Divide la cadena en una lista de subcadenas, utilizando `sep` como la cadena delimitadora, comenzando desde el extremo derecho de la cadena. Si `maxsplit` se especifica y es no negativo, se realizan como máximo `maxsplit` divisiones, afectando primero a los elementos más a la derecha. Si `maxsplit` no se especifica o es `-1`, no hay límite en el número de divisiones (se realizan todas las divisiones posibles).

Si se especifica `sep`, se utiliza como delimitador entre subcadenas, y los delimitadores consecutivos darán como resultado subcadenas vacías (por ejemplo, `'1,,2'.rsplit(',', maxsplit=1)` devuelve `['1,', '2']`). Si `sep` no se especifica o es `None`, cualquier secuencia de caracteres en blanco consecutivos se trata como un separador. Este comportamiento es similar a `split()`, pero la división comienza desde el final de la cadena y se mueve hacia la izquierda.

Ejemplo

```
'abc,def,ghi,jkl'.rsplit(',',2) # ['abc,def', 'ghi', 'jkl']
```

1867 startswith (builtins.str) — Devuelve `True` si la cadena comienza con el prefijo especificado ``prefix``, de lo contrario devuelve `False`.

ID 1867 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `startswith (builtins.str)`

Explicación española: Devuelve `True` si la cadena comienza con el prefijo especificado ``prefix``, de lo contrario devuelve `False`.

Sintaxis

```
startswith(prefix [, start=0])
```

Devuelve `True` si la cadena comienza con el prefijo especificado `prefix`, de lo contrario devuelve `False`. Si se proporciona el parámetro opcional `start`, la prueba comienza en esa posición dentro de la cadena.

Ejemplo

```
'Hello World'.startswith('He') # True
```

1868 endswith (builtins.str) — Devuelve `True` si la cadena termina con el sufijo especificado ``suffix``, de lo contrario devuelve `False`.

ID 1868 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `endswith (builtins.str)`

Explicación española: Devuelve `True` si la cadena termina con el sufijo especificado ``suffix``, de lo contrario devuelve `False`.

Sintaxis

```
endswith(suffix)
```

Devuelve `True` si la cadena termina con el sufijo especificado `suffix`, de lo contrario devuelve `False`. Este método verifica si la cadena concluye con el carácter o la secuencia de caracteres especificados en el sufijo.

1869 strip (builtins.str) — Devuelve una copia de la cadena con los caracteres iniciales y finales especificados eliminados.

ID 1869 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `strip (builtins.str)`

Explicación española: Devuelve una copia de la cadena con los caracteres iniciales y finales especificados eliminados.

Sintaxis

```
strip([chars])
```

Devuelve una copia de la cadena con los caracteres iniciales y finales especificados eliminados. El argumento `chars` es una cadena que especifica el conjunto de caracteres a eliminar. Si se omite o es `None`, el argumento `chars` por defecto elimina espacios en blanco ASCII. El argumento `chars` no es un prefijo o sufijo; más bien, todas las combinaciones de sus valores se eliminan tanto del principio como del final de la cadena.

Ejemplos

```
' hello '.strip() # 'hello'
'www.example.com'.strip('w.moc') # 'example'
```

1870 lstrip (builtins.str) — Devuelve una copia de la cadena con los caracteres iniciales especificados eliminados.

ID 1870 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `lstrip (builtins.str)`

Explicación española: Devuelve una copia de la cadena con los caracteres iniciales especificados eliminados.

Sintaxis

```
lstrip([chars])
```

Devuelve una copia de la cadena con los caracteres iniciales especificados eliminados. El argumento `chars` es una cadena que especifica el conjunto de caracteres a eliminar del principio de la cadena. Si se omite o es `None`, el argumento `chars` por defecto elimina espacios en blanco ASCII iniciales. El argumento `chars` no es un prefijo; más bien, todas las combinaciones de sus valores se eliminan del inicio de la cadena.

Ejemplo

```
***hello***.lstrip('*') # 'hello**'
```

1871 rstrip (builtins.str) — Devuelve una copia de la cadena con los caracteres finales especificados eliminados.

ID 1871 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `rstrip (builtins.str)`

Explicación española: Devuelve una copia de la cadena con los caracteres finales especificados eliminados.

Sintaxis

```
rstrip([chars])
```

Devuelve una copia de la cadena con los caracteres finales especificados eliminados. El argumento `chars` es una cadena que especifica el conjunto de caracteres a eliminar del final de la cadena. Si se omite o es `None`, el argumento `chars` por defecto elimina espacios en blanco ASCII finales. El argumento `chars` no es un sufijo; más bien, todas las combinaciones de sus valores se eliminan del final de la cadena.

Ejemplo

```
***hello***.rstrip('*') # '***hello'
```

1872 format (builtins.str) — Realiza una operación de formateo de cadenas.

ID 1872 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `format (builtins.str)`

Explicación española: Realiza una operación de formateo de cadenas.

Sintaxis

```
format(*args, **kwargs)
```

Realiza una operación de formateo de cadenas. La cadena sobre la que se llama a este método puede contener texto literal y campos de reemplazo delimitados por llaves `{}`. Las llaves que deban mostrarse como texto literal deben duplicarse: `{{ y }}`.

Cada campo de reemplazo contiene el índice numérico de un argumento posicional o el nombre de un argumento de palabra clave. No se admite el acceso a atributos ni a elementos indexados dentro de un campo de reemplazo.

Devuelve una copia de la cadena en la que cada campo de reemplazo se sustituye por el valor formateado del argumento correspondiente.

Sintaxis de la especificación de formato:

```
{[nombre_campo]![conversión]:[[relleno]alineación][signo][#][0][ancho][.][precisión][tipo]}
```

Especificadores de tipo `[tipo]`:

- Para enteros: d: representación decimal b: representación binaria (base 2) o: representación octal (base 8) x: representación hexadecimal en minúsculas (base 16) X: representación hexadecimal en mayúsculas (base 16) c: carácter Unicode n: se acepta y se comporta como la representación predeterminada de enteros
- Para números de punto flotante: f: notación de punto fijo; la precisión predeterminada es de 6 decimales F: igual que f, pero muestra INF y NAN en mayúsculas e: notación exponencial con e minúscula, por ejemplo 1.23e+02 E: notación exponencial con E mayúscula, por ejemplo 1.23E+02 g: formato general que utiliza notación de punto fijo o exponencial, según el valor y la precisión G: igual que g, pero utiliza letras mayúsculas cuando corresponde %: formato de porcentaje; multiplica el valor por 100 y añade % n: se acepta y se comporta como la representación predeterminada de números de punto flotante

- Para cadenas: s: representación de cadena

Si se omite [tipo], se utiliza el formato predeterminado del valor.

Alineación [alineación]:

- <: alinear a la izquierda
- >: alinear a la derecha
- ^: centrar
- =: insertar el relleno después del signo; solo para tipos numéricos

Carácter de relleno [relleno]:

- Puede utilizarse cualquier carácter como relleno. El carácter de relleno predeterminado es un espacio.

Signo [signo]:

- +: mostrar siempre el signo
- -: mostrar el signo solo para números negativos; esta es la opción predeterminada
- (espacio): mostrar un espacio inicial para números positivos y un signo menos para números negativos

Forma alternativa [#]:

- #: activa la forma alternativa para tipos enteros: Binario (b): añade el prefijo 0b Octal (o): añade el prefijo 0o Hexadecimal (x, X): añade el prefijo 0x o 0X

En los formatos de punto flotante, # no fuerza la inclusión de un punto decimal ni conserva los ceros finales en esta implementación.

Indicadores de conversión [!conversión]:

- !s: convertir mediante str()
- !r: convertir mediante repr()

Separador de agrupación [,]:

- ,: añade separadores de agrupación a los enteros

En los enteros decimales, los dígitos se agrupan de tres en tres. En los enteros binarios, los dígitos también pueden agruparse de tres en tres. No se aplica agrupación a los números de punto flotante. El guion bajo _ no se admite como separador de agrupación.

Ancho [ancho] y precisión [.precisión]:

- ancho: ancho mínimo del campo
- .precisión: número de decimales en formatos de punto fijo, número de cifras significativas en formatos generales o longitud máxima en cadenas

Pueden utilizarse campos de reemplazo anidados para especificar dinámicamente el ancho y la precisión.

La numeración automática y manual de campos no puede combinarse en la misma cadena de formato.

Ejemplos

```
"La suma de 1 + 2 es {0}".format(1+2)
# 'La suma de 1 + 2 es 3'
```

```
"Valores: {0}, {1}, {2}".format(10, 20, 30)
# 'Valores: 10, 20, 30'
```

```
'Hola, {nombre}!'.format(nombre='Juan')
# 'Hola, Juan!'
```

```
"Centro: '{0:^10}'".format('test')
# "Centro: ' test '"
```

```
"Ceros iniciales: {0:010d}".format(15)
# 'Ceros iniciales: 0000000015'
```

```
"HEX: {0:X} vs {0:#X}".format(255)
# 'HEX: FF vs 0XFF'

"Moneda: ${0:.2f}".format(1234.5678)
# 'Moneda: $1234.57'
```

1873 replace (builtins.str) — Devuelve una copia de la cadena con todas las ocurrencias de la subcadena `old` reemplazadas por `new`.

ID 1873 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `replace (builtins.str)`

Explicación española: Devuelve una copia de la cadena con todas las ocurrencias de la subcadena `old` reemplazadas por `new`.

Sintaxis

```
replace(old, new [, count])
```

Devuelve una copia de la cadena con todas las ocurrencias de la subcadena `old` reemplazadas por `new`. Si se proporciona el argumento opcional `count`, solo se reemplazan las primeras `count` ocurrencias.

Nota: Este método produce un nuevo objeto de cadena y no modifica la cadena original en su lugar.

1877 center (builtins.str) — Devuelve una copia de la cadena centrada en una nueva cadena de longitud ``width``.

ID 1877 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `center (builtins.str)`

Explicación española: Devuelve una copia de la cadena centrada en una nueva cadena de longitud ``width``.

Sintaxis

```
center(width)
```

Devuelve una copia de la cadena centrada en una nueva cadena de longitud `width`. El relleno se realiza utilizando el carácter de espacio. Si `width` es menor o igual a la longitud de la cadena original, se devuelve la cadena original.

1874 count (builtins.str) — Devuelve el número de ocurrencias no superpuestas de la subcadena ``sub`` en el rango `[`start`, `end`]`.

ID 1874 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `count (builtins.str)`

Explicación española: Devuelve el número de ocurrencias no superpuestas de la subcadena ``sub`` en el rango `[`start`, `end`]`.

Sintaxis

```
count(sub [, start [, end]])
```

Devuelve el número de ocurrencias no superpuestas de la subcadena `sub` en el rango `[start, end]`. Los argumentos opcionales `start` y `end` se interpretan como en la notación de rebanado. Este método cuenta cuántas veces aparece la subcadena especificada dentro de un segmento específico de la cadena principal.

1875 partition (builtins.str) — Divide la cadena en la primera aparición del separador ``sep`` y devuelve una tupla de 3 elementos que...

ID 1875 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `partition (builtins.str)`

Explicación española: Divide la cadena en la primera aparición del separador ``sep`` y devuelve una tupla de 3 elementos que...

Sintaxis

```
partition(sep)
```

Divide la cadena en la primera aparición del separador `sep` y devuelve una tupla de 3 elementos que contiene la parte antes del separador, el separador en sí y la parte después del separador. Si no se encuentra el separador, el método devuelve una tupla de 3 elementos que contiene la cadena original, seguida de dos cadenas vacías.

El separador `sep` a buscar puede ser cualquier objeto de cadena.

1876 `rpartition` (`builtins.str`) — Divide la cadena en la última ocurrencia del separador ``sep`` y devuelve una tupla de 3 elementos que...

ID 1876 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `rpartition` (`builtins.str`)

Explicación española: Divide la cadena en la última ocurrencia del separador ``sep`` y devuelve una tupla de 3 elementos que...

Sintaxis

`rpartition(sep)`

Divide la cadena en la última ocurrencia del separador `sep` y devuelve una tupla de 3 elementos que contiene la parte antes del separador, el separador en sí y la parte después del separador. Si no se encuentra el separador, el método devuelve una tupla de 3 elementos que contiene dos cadenas vacías, seguidas de una copia de la cadena original.

El separador `sep` a buscar puede ser cualquier objeto de cadena.

1878 `lower` (`builtins.str`) — Devuelve una copia de la cadena con todos los caracteres ASCII en mayúsculas convertidos a sus...

ID 1878 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `lower` (`builtins.str`)

Explicación española: Devuelve una copia de la cadena con todos los caracteres ASCII en mayúsculas convertidos a sus...

Sintaxis

`lower()`

Devuelve una copia de la cadena con todos los caracteres ASCII en mayúsculas convertidos a sus equivalentes en minúsculas correspondientes.

1879 `upper` (`builtins.str`) — Devuelve una copia de la cadena con todos los caracteres ASCII en minúsculas convertidos a su...

ID 1879 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `upper` (`builtins.str`)

Explicación española: Devuelve una copia de la cadena con todos los caracteres ASCII en minúsculas convertidos a su...

Sintaxis

`upper()`

Devuelve una copia de la cadena con todos los caracteres ASCII en minúsculas convertidos a su correspondiente equivalente en mayúsculas.

1880 `isspace` (`builtins.str`) — Devuelve `True` si todos los caracteres de la cadena son espacios en blanco ASCII y la cadena no está...

ID 1880 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `isspace` (`builtins.str`)

Explicación española: Devuelve `True` si todos los caracteres de la cadena son espacios en blanco ASCII y la cadena no está...

Sintaxis

`isspace()`

Devuelve True si todos los caracteres de la cadena son espacios en blanco ASCII y la cadena no está vacía; de lo contrario, devuelve False. Los caracteres de espacio en blanco ASCII incluyen espacio (), tabulación (\t), nueva línea (\n), retorno de carro (\r), tabulación vertical (\v) y avance de página (\f).

1881 isalpha (builtins.str) — Devuelve True si todos los caracteres de la cadena son caracteres ASCII alfabéticos y la cadena no está...

ID 1881 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `isalpha (builtins.str)`

Explicación española: Devuelve True si todos los caracteres de la cadena son caracteres ASCII alfabéticos y la cadena no está...

Sintaxis

`isalpha()`

Devuelve True si todos los caracteres de la cadena son caracteres ASCII alfabéticos y la cadena no está vacía; de lo contrario, devuelve False. Los caracteres ASCII alfabéticos son aquellos en los rangos 'a' a 'z' y 'A' a 'Z'.

1882 isdigit (builtins.str) — Devuelve True si todos los caracteres de la cadena son dígitos decimales ASCII y la cadena no está...

ID 1882 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `isdigit (builtins.str)`

Explicación española: Devuelve True si todos los caracteres de la cadena son dígitos decimales ASCII y la cadena no está...

Sintaxis

`isdigit()`

Devuelve True si todos los caracteres de la cadena son dígitos decimales ASCII y la cadena no está vacía; de lo contrario, devuelve False. Los dígitos decimales ASCII incluyen los caracteres de '0' a '9'.

1883 isupper (builtins.str) — Devuelve True si hay al menos un carácter alfabético ASCII en mayúsculas en la cadena y no hay...

ID 1883 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `isupper (builtins.str)`

Explicación española: Devuelve True si hay al menos un carácter alfabético ASCII en mayúsculas en la cadena y no hay...

Sintaxis

`isupper()`

Devuelve True si hay al menos un carácter alfabético ASCII en mayúsculas en la cadena y no hay caracteres ASCII en minúsculas; de lo contrario, devuelve False. Este método verifica si la cadena contiene exclusivamente letras mayúsculas, sin letras minúsculas.

1884 islower (builtins.str) — Devuelve True si hay al menos un carácter alfabético ASCII en minúscula en la cadena y no hay caracteres...

ID 1884 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `islower (builtins.str)`

Explicación española: Devuelve True si hay al menos un carácter alfabético ASCII en minúscula en la cadena y no hay caracteres...

Sintaxis

`islower()`

Devuelve True si hay al menos un carácter alfabético ASCII en minúscula en la cadena y no hay caracteres ASCII en mayúscula; de lo contrario, devuelve False. Este método verifica si la cadena contiene exclusivamente letras minúsculas, sin letras mayúsculas.

1885 super (builtins) — Devuelve un objeto proxy que delega las llamadas a métodos a una clase padre o hermana de tipo.

ID 1885 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `super` (builtins)

Explicación española: Devuelve un objeto proxy que delega las llamadas a métodos a una clase padre o hermana de tipo.

Sintaxis

```
super(type, object)
super(type, type)
```

Devuelve un objeto proxy que delega las llamadas a métodos a una clase padre o hermana de tipo. Esto es útil para acceder a métodos heredados que han sido anulados en una clase.

El objeto o tipo determina el orden de resolución de métodos a buscar. La búsqueda comienza desde la clase justo después del tipo.

Por ejemplo, si el mro de object-or-type es $D \rightarrow B \rightarrow C \rightarrow A \rightarrow \text{object}$ y el valor de type es B, entonces `super()` busca $C \rightarrow A \rightarrow \text{object}$.

El atributo mro del object-or-type enumera el orden de búsqueda de resolución de métodos utilizado por `getattr()` y `super()`. El atributo es dinámico y puede cambiar cada vez que se actualiza la jerarquía de herencia.

Si se omite el segundo argumento, el objeto `super` devuelto es no vinculado. Si el segundo argumento es un objeto, `isinstance(obj, type)` debe ser verdadero. Si el segundo argumento es un tipo, `issubclass(type2, type)` debe ser verdadero (esto es útil para métodos de clase).

Hay dos casos de uso típicos para `super`. En una jerarquía de clases con herencia simple, `super` se puede usar para referirse a las clases padre sin nombrarlas explícitamente, lo que hace que el código sea más mantenible. Este uso se asemeja al uso de `super` en otros lenguajes de programación.

El segundo caso de uso es para admitir la herencia múltiple cooperativa en un entorno de ejecución dinámica. Este caso de uso es único de Python y no se encuentra en lenguajes compilados estáticamente o lenguajes que solo admiten herencia simple. Esto hace posible implementar "diagramas de diamante" donde múltiples clases base implementan el mismo método. Un buen diseño dicta que este método tenga la misma firma de llamada en cada caso (porque el orden de las llamadas se determina en tiempo de ejecución, porque ese orden se adapta a los cambios en la jerarquía de clases y porque ese orden puede incluir clases hermanas que son desconocidas antes del tiempo de ejecución).

Para ambos casos de uso, una llamada típica a la superclase se ve así:

```
class C(B):
    def method(self, arg):
        super().method(arg) # Esto hace lo mismo que super(C, self).method(arg)
```

Además de las búsquedas de métodos, `super()` también funciona para búsquedas de atributos. Un posible caso de uso para esto es llamar a descriptores en una clase padre o hermana.

Tenga en cuenta que `super()` se implementa como parte del proceso de vinculación para búsquedas de atributos explícitos con punto, como `super().getitem(name)`. Lo hace implementando su propio método `getattribute()` para buscar clases en un orden predecible que admite la herencia múltiple cooperativa. En consecuencia, `super()` no está definido para búsquedas implícitas utilizando declaraciones u operadores como `super()[name]`.

También tenga en cuenta que, aparte de la forma de cero argumentos, `super()` no está limitado a su uso dentro de métodos. La forma de dos argumentos especifica los argumentos exactamente y hace las referencias apropiadas. La forma de cero argumentos solo funciona dentro de una definición de clase, ya que el compilador completa los detalles necesarios para recuperar correctamente la clase que se está definiendo, así como para acceder a la instancia actual para métodos ordinarios.

1886 tuple, clase (builtins)

ID 1886 · título localizado en HelpEs

Sintaxis

```
tuple([i])
```

Convierte el iterable `i` en una tupla. Si no se proporciona un iterable, crea una tupla vacía.

La clase `tuple` en Python representa una secuencia inmutable de elementos. A diferencia de las listas, las tuplas no se pueden modificar después de su creación, lo que las hace adecuadas para almacenar colecciones fijas de elementos. Las tuplas son ordenadas y pueden contener elementos de diferentes tipos, incluidos otros tuples. A menudo se utilizan para colecciones de tamaño fijo o como claves de diccionario.

Ejemplo

```
tuple([1,2,3]) # (1,2,3)
tuple('abc') # ('a','b','c')
tuple({'a':1,'b':2}) # ('a','b')
```

1887 count (builtins.tuple) — Devuelve el número de ocurrencias del elemento `value` en la tupla.

ID 1887 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `count (builtins.tuple)`

Explicación española: Devuelve el número de ocurrencias del elemento `value` en la tupla.

Sintaxis

`count(value)`

Devuelve el número de ocurrencias del elemento `value` en la tupla. El elemento `value` puede ser de cualquier tipo, incluidas otras tuplas. Cuando `value` es una tupla u otro objeto complejo, el método verifica si hay una coincidencia exacta, lo que significa que solo cuenta las instancias donde `value` es idéntico a un objeto específico en la tupla (no solo que contenga los mismos elementos).

Ejemplo

```
(1, 2, 3, 2, 4, 2, 5).count(2) # 3
(1, (2, 3), 3, 4, (2, 3)).count(3) # 1
```

1888 index (builtins.tuple) — Devuelve el índice de la primera ocurrencia del elemento `value` en la tupla.

ID 1888 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `index (builtins.tuple)`

Explicación española: Devuelve el índice de la primera ocurrencia del elemento `value` en la tupla.

Sintaxis

`index(value [, start [, end]])`

Devuelve el índice de la primera ocurrencia del elemento `value` en la tupla. La búsqueda se puede restringir a un rango específico dentro de la tupla utilizando los argumentos opcionales `start` y `end`. Si no se encuentra el elemento, se genera un `ValueError`. El valor del elemento puede ser de cualquier tipo, incluidas otras tuplas. El método busca una coincidencia exacta cuando `value` es un objeto complejo, como otra tupla.

Ejemplo

```
(1, 2, 3, 2, 4, 2, 5).index(2) # 1
(1, 2, 3, 2, 4, 2, 5).index(2,2) # 3
```

1889 type class (builtins) — La clase incorporada `type` en Python cumple dos propósitos principales: identifica la clase de un objeto...

ID 1889 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `type class (builtins)`

Explicación española: La clase incorporada `type` en Python cumple dos propósitos principales: identifica la clase de un objeto...

Sintaxis

`type(object)`
`type(name, bases, dict)`

La clase incorporada `type` en Python cumple dos propósitos principales: identifica la clase de un objeto existente y crea nuevas clases dinámicamente.

- Identificación e introspección de tipos: `type(object)`: Devuelve la clase del objeto especificado. Esta forma se utiliza habitualmente para inspeccionar el tipo de un objeto en tiempo de ejecución.
- Creación dinámica de tipos: `type(name, bases, dict)`: Crea y devuelve una nueva clase dinámicamente. `name` especifica el nombre de la nueva clase. `bases` es una tupla que contiene las clases de las que hereda la nueva clase. `dict` es un diccionario que contiene los atributos y métodos asignados a la nueva clase.

1890 `zip (builtins)` — Crea un iterador que agrega elementos de cada uno de los iterables.

ID 1890 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `zip (builtins)`

Explicación española: Crea un iterador que agrega elementos de cada uno de los iterables.

Sintaxis

```
zip(*iterables)
```

Crea un iterador que agrega elementos de cada uno de los iterables.

Devuelve un iterador de tuplas, donde la *i*-ésima tupla contiene el *i*-ésimo elemento de cada una de las secuencias o iterables de argumentos. El iterador se detiene cuando el iterable de entrada más corto se agota. Con un solo argumento iterable, devuelve un iterador de tuplas de 1. Sin argumentos, devuelve un iterador vacío.

1891 `classmethod (builtins)` — Transforma un método en un método de clase.

ID 1891 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `classmethod (builtins)`

Explicación española: Transforma un método en un método de clase.

Sintaxis

```
@classmethod
```

Transforma un método en un método de clase.

Un método de clase recibe la clase como primer argumento implícito, al igual que un método de instancia recibe la instancia. Para declarar un método de clase, use este idioma:

```
class C:
    @classmethod
    def f(cls, arg1, arg2, ...): ...
```

La forma `@classmethod` es un decorador de funciones – vea Definiciones de funciones para más detalles.

Un método de clase puede ser llamado tanto en la clase (como `C.f()`) como en una instancia (como `C().f()`). La instancia se ignora excepto por su clase. Si se llama a un método de clase para una clase derivada, el objeto de la clase derivada se pasa como el primer argumento implícito.

1892 `staticmethod (builtins)` — Transforma un método en un método estático.

ID 1892 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `staticmethod (builtins)`

Explicación española: Transforma un método en un método estático.

Sintaxis

```
@staticmethod
```

Transforma un método en un método estático.

Un método estático no recibe un primer argumento implícito. Para declarar un método estático, utilice este modismo:

```
class C:
    @staticmethod
    def f(arg1, arg2, ...): ...
```

La forma `@staticmethod` es un decorador de función - consulta Definiciones de funciones para más detalles.

Un método estático se puede llamar tanto en la clase (como `C.f()`) como en una instancia (como `C().f()`).

Los métodos estáticos en Python son similares a los que se encuentran en Java o C++. También consulta `classmethod()` para una variante que es útil para crear constructores de clase alternativos.

1895 abs (builtins) — Devuelve el valor absoluto de un número.

ID 1895 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>abs (builtins)</code> Explicación española: Devuelve el valor absoluto de un número.
Sintaxis <code>abs(x)</code>

Devuelve el valor absoluto de un número. El argumento puede ser un número entero o un número de punto flotante. Si el argumento es un número complejo, se devuelve su magnitud. Si `x` define `abs()`, `abs(x)` devuelve `x.abs()`.

1896 all (builtins) — Devuelve True si todos los elementos del iterable `i` son verdaderos, o si `i` está vacío.

ID 1896 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>all (builtins)</code> Explicación española: Devuelve True si todos los elementos del iterable <code>i</code> son verdaderos, o si <code>i</code> está vacío.
Sintaxis <code>all(i)</code>

Devuelve True si todos los elementos del iterable `i` son verdaderos, o si `i` está vacío.

1897 any (builtins) — Devuelve True si algún elemento del iterable `i` es verdadero.

ID 1897 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>any (builtins)</code> Explicación española: Devuelve True si algún elemento del iterable <code>i</code> es verdadero.
Sintaxis <code>any(i)</code>

Devuelve True si algún elemento del iterable `i` es verdadero.

1898 bin (builtins) — Convierte un número entero `x` en una cadena binaria con el prefijo `"0b"`.

ID 1898 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>bin (builtins)</code> Explicación española: Convierte un número entero <code>x</code> en una cadena binaria con el prefijo <code>"0b"</code> .
Sintaxis <code>bin(x)</code>

Convierte un número entero `x` en una cadena binaria con el prefijo `"0b"`. El resultado es una expresión válida de Python. Si `x` no es un objeto `int` de Python, debe definir un método `index()` que devuelva un entero.

1899 callable (builtins) — Devuelve True si el objeto `obj` parece ser llamable, False si no.

ID 1899 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>callable (builtins)</code> Explicación española: Devuelve True si el objeto <code>obj</code> parece ser llamable, False si no.

Sintaxis

`callable(obj)`

Devuelve True si el objeto `obj` parece ser llamable, False si no. Si esto devuelve True, aún es posible que una llamada falle, pero si es False, la llamada al objeto nunca tendrá éxito. Tenga en cuenta que las clases son llamables (llamar a una clase devuelve una nueva instancia); las instancias son llamables si su clase tiene un método `call()`.

1900 compile (builtins) — Compila el código fuente en un objeto de código.

ID 1900 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `compile` (builtins)

Explicación española: Compila el código fuente en un objeto de código.

Sintaxis

`compile(source, filename, mode)`

Compila el código fuente en un objeto de código. Los objetos de código pueden ser ejecutados por `exec()` o `eval()`.

El argumento `source` es el código fuente a compilar. Puede ser una cadena normal o una cadena de bytes.

El argumento `filename` debe indicar el archivo del que se leyó el código; proporcione un valor reconocible si no se leyó de un archivo ('<string>' se usa comúnmente).

El argumento `mode` especifica qué tipo de código debe ser compilado; puede ser 'exec' si la fuente consiste en una secuencia de declaraciones, 'eval' si consiste en una sola expresión, o 'single' si consiste en una sola declaración interactiva (en este último caso, las declaraciones de expresión que evalúan a algo diferente de None se imprimirán).

Ejemplo

```
code = compile('3 * x', '', 'eval')
eval(code, {'x': 2}) # salida: 6
```

1901 chr (builtins) — Devuelve la cadena que representa un carácter cuyo punto de código Unicode es el entero `i`.

ID 1901 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `chr` (builtins)

Explicación española: Devuelve la cadena que representa un carácter cuyo punto de código Unicode es el entero `i`.

Sintaxis

`chr(i)`

Devuelve la cadena que representa un carácter cuyo punto de código Unicode es el entero `i`. Por ejemplo, `chr(97)` devuelve la cadena 'a', mientras que `chr(8364)` devuelve la cadena '€'. Esta es la inversa de `ord()`.

El rango válido para el argumento es de 0 a 1,114,111 (0x10FFFF en base 16). Se generará un `ValueError` si `i` está fuera de ese rango.

1902 delattr (builtins) — Este es un pariente de `setattr()`.

ID 1902 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `delattr` (builtins)

Explicación española: Este es un pariente de `setattr()`.

Sintaxis

`delattr(obj, name)`

Este es un pariente de `setattr()`. Los argumentos son un objeto `obj` y una cadena `name`. La cadena debe ser el nombre de uno de los atributos del objeto. La función elimina el atributo nombrado, siempre que el objeto lo permita. Por ejemplo, `delattr(x, 'foobar')` es equivalente a `del x.foobar`.

1903 dir (builtins) — Sin argumentos, devuelve la lista de nombres en el ámbito local actual.

Referencia traducidaIdentificador técnico: `dir` (builtins)

Explicación española: Sin argumentos, devuelve la lista de nombres en el ámbito local actual.

Sintaxis`dir([obj])`

Sin argumentos, devuelve la lista de nombres en el ámbito local actual. Con un argumento `obj`, intenta devolver una lista de atributos válidos para el objeto `obj`.

Si el objeto tiene un método llamado `dir()`, se llamará a este método y debe devolver la lista de atributos. Esto permite que los objetos que implementan una función personalizada `getattr()` o `getattribute()` personalicen la forma en que `dir()` informa sobre sus atributos.

Si el objeto no proporciona `dir()`, la función intenta recopilar información del atributo `dict` del objeto, si está definido, y de su objeto de tipo. La lista resultante no es necesariamente completa y puede ser inexacta cuando el objeto tiene un `getattr()` personalizado.

El mecanismo `dir()` por defecto se comporta de manera diferente con diferentes tipos de objetos, ya que intenta producir la información más relevante, en lugar de completa:

Si el objeto es un objeto módulo, la lista contiene los nombres de los atributos del módulo.

Si el objeto es un objeto tipo o clase, la lista contiene los nombres de sus atributos y, de manera recursiva, los atributos de sus bases.

De lo contrario, la lista contiene los nombres de los atributos del objeto, los nombres de los atributos de su clase y, de manera recursiva, los atributos de las clases base de su clase.

1904 divmod (builtins) — Toma dos números (no complejos) ``a`` y ``b`` como argumentos y devuelve un par de números que consiste en...

Referencia traducidaIdentificador técnico: `divmod` (builtins)Explicación española: Toma dos números (no complejos) ``a`` y ``b`` como argumentos y devuelve un par de números que consiste en...**Sintaxis**`divmod(a, b)`

Toma dos números (no complejos) `a` y `b` como argumentos y devuelve un par de números que consiste en su cociente y residuo al usar la división entera. Con tipos de operandos mixtos, se aplican las reglas para los operadores aritméticos binarios. Para enteros, el resultado es el mismo que `(a // b, a % b)`. Para números de punto flotante, el resultado es `(q, a % b)`, donde `q` es generalmente `math.floor(a / b)` pero puede ser 1 menos que eso. En cualquier caso, `q * b + a % b` es muy cercano a `a`, si `a % b` no es cero, tiene el mismo signo que `b`, y `0 <= abs(a % b) < abs(b)`.

1905 eval (builtins) — ``expression``: una cadena que es la expresión de Python a evaluar.

Referencia traducidaIdentificador técnico: `eval` (builtins)Explicación española: ``expression``: una cadena que es la expresión de Python a evaluar.**Sintaxis**`eval(expression [, globals [, locals]])`**Parámetros**

- `expression`: una cadena que es la expresión de Python a evaluar.
- `globals` (opcional): un diccionario que contiene variables globales accesibles en `expression`. Si se proporciona, debe ser un diccionario.
- `locals` (opcional): cualquier objeto de mapeo que proporcione variables locales. Esto puede ser cualquier tipo de mapeo, no estrictamente un diccionario, utilizado para restringir aún más el espacio de nombres para la expresión.

El argumento `expression` se analiza y evalúa como una expresión de Python (técnicamente hablando, una lista condicional) utilizando los diccionarios `globals` y `locals` como los espacios de nombres global y local, respectivamente. Si se proporciona el diccionario `globals` y no contiene una clave para `builtins`, se inserta automáticamente una referencia al diccionario del módulo incorporado `builtins` bajo esa clave antes de que se analice la expresión. Esto asegura que la expresión normalmente tenga acceso completo al módulo estándar de funciones incorporadas, y que cualquier entorno restringido se propague correctamente.

Si se omite el diccionario `locals`, por defecto será el diccionario `globals`. Si se omiten ambos diccionarios, `globals` y `locals`, la expresión se ejecuta dentro de los espacios de nombres global y local donde se llama a la función `eval()`. Tenga en cuenta que `eval()` no tiene acceso a los ámbitos anidados (no locales) en el entorno circundante.

El valor de retorno de `eval()` es el resultado de la expresión evaluada. Cualquier error de sintaxis en la expresión se informa como excepción.

Esta función también se puede usar para ejecutar objetos de código arbitrarios (por ejemplo, aquellos creados por la función `compile()`). En tales casos, se debe pasar un objeto de código en lugar de una cadena. Si el objeto de código se ha compilado con `exec` como argumento de modo, el valor de retorno de `eval()` será `None`.

Nota: Es esencial comprender los roles de `globals` y `locals` en el contexto de `eval()`:

- `globals`: Esto debe ser un diccionario que represente el entorno global en el que se debe evaluar la expresión. Afecta todo el acceso a variables globales dentro de la expresión.
- `locals`: Esto se usa para definir el entorno local para la expresión. Puede anular los `globals` para variables específicas accesibles en la expresión.

Ejemplo

```
expr_1 = '3 + 4'
result_1 = eval(expr_1)
print(result_1) # Salida: 7
```

```
environment = {'x': 10, 'y': 20}
expr_2 = 'x * y + 5'
result_2 = eval(expr_2, environment)
print(result_2) # Salida: 205
```

1906 `exec (builtins)` — Descripción: Esta función permite la ejecución dinámica de código Python.

ID 1906 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `exec (builtins)`

Explicación española: Descripción: Esta función permite la ejecución dinámica de código Python.

Sintaxis

```
exec(object [, globals [, locals]])
```

Descripción:

Esta función permite la ejecución dinámica de código Python. El argumento `object` debe ser una cadena o un objeto de código:

- Si `object` es una cadena, se analiza como un conjunto de declaraciones de Python que luego se ejecutan, a menos que ocurra un error de sintaxis.
- Si `object` es un objeto de código, se ejecuta directamente. En todos los casos, se espera que el código ejecutado sea válido como entrada de archivo `*.py`, siguiendo las reglas de sintaxis y estructura de los archivos Python. Tenga en cuenta que las declaraciones `'return'` y `'yield'` están prohibidas fuera de las definiciones de funciones, incluso cuando se utilizan dentro del código pasado a la función `exec()`. El valor de retorno de `exec()` siempre es `None`.

Manejo del ámbito:

- Si no se proporcionan argumentos opcionales, el código se ejecuta en el ámbito actual del llamador.
- Si solo se proporciona `globals`, debe ser un diccionario (no una subclase de diccionario). Este diccionario servirá como el entorno global y local para el código ejecutado.
- Si se proporcionan tanto `globals` como `locals`, se utilizarán para definir los entornos global y local, respectivamente.
- `locals` puede ser cualquier objeto de mapeo, no solo diccionarios. A nivel de módulo, `globals` y `locals` generalmente se refieren al mismo diccionario.

Uso de `globals` y `locals`:

- **globals:** Este diccionario define el entorno global en el que se ejecuta el código. Debe contener todas las variables y funciones globales necesarias para el código. Si el diccionario `globals` no incluye una clave para `builtins`, se inserta automáticamente una referencia al diccionario del módulo incorporado bajo esa clave.
- **locals:** Este objeto de mapeo define el entorno local para el código ejecutado. Puede anular los valores en el diccionario `globals` durante el alcance del código ejecutado.

Ejemplo

```
# Definir el diccionario `globals`
globals_dict = {'x': 5, 'y': 10}
# Código a ejecutar
code_to_execute = 'result = x + y'
# Ejecutar el código usando exec(), pasando globals_dict
exec(code_to_execute, globals_dict)
# Imprimir el resultado de los cálculos
print(globals_dict['result']) # Salida: 15
```

1907 `getattr (builtins)` — Devuelve el valor del atributo nombrado del objeto ``obj``.

ID 1907 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `getattr (builtins)`

Explicación española: Devuelve el valor del atributo nombrado del objeto ``obj``.

Sintaxis

```
getattr(obj, name [, default])
```

Devuelve el valor del atributo nombrado del objeto `obj`. El `name` debe ser una cadena. Si la cadena es el nombre de uno de los atributos del objeto, el resultado es el valor de ese atributo. Por ejemplo, `getattr(x, 'foobar')` es equivalente a `x.foobar`. Si el atributo nombrado no existe, se devuelve `default` si se proporciona, de lo contrario se genera un `AttributeError`.

1908 `setattr (builtins)` — Este es el equivalente de `getattr()`.

ID 1908 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `setattr (builtins)`

Explicación española: Este es el equivalente de `getattr()`.

Sintaxis

```
setattr(obj, name, value)
```

Este es el equivalente de `getattr()`. Los argumentos son un objeto `obj`, una cadena `name` y un valor arbitrario. La cadena puede nombrar un atributo existente o un nuevo atributo. La función asigna el `value` al atributo, siempre que el objeto lo permita. Por ejemplo, `setattr(x, 'foobar', 123)` es equivalente a `x.foobar = 123`.

1909 `globals (builtins)` — Devuelve un diccionario que representa la tabla de símbolos global actual.

ID 1909 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `globals (builtins)`

Explicación española: Devuelve un diccionario que representa la tabla de símbolos global actual.

Sintaxis

```
globals()
```

Devuelve un diccionario que representa la tabla de símbolos global actual. Este es siempre el diccionario del módulo actual (dentro de una función o método, este es el módulo donde se define, no el módulo desde el que se llama).

1910 `hasattr (builtins)` — Los argumentos son un objeto y una cadena.

ID 1910 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `hasattr` (builtins)

Explicación española: Los argumentos son un objeto y una cadena.

Sintaxis

`hasattr(objeto, nombre)`

Los argumentos son un objeto y una cadena. El resultado es `True` si la cadena es el nombre de uno de los atributos del objeto, `False` si no. (Esto se implementa llamando a `getattr(objeto, nombre)` y viendo si se genera un `AttributeError` o no.)

1911 hash (builtins) — Devuelve el valor hash del objeto (si tiene uno).

ID 1911 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `hash` (builtins)

Explicación española: Devuelve el valor hash del objeto (si tiene uno).

Sintaxis

`hash(objeto)`

Devuelve el valor hash del objeto (si tiene uno). Los valores hash son enteros. Se utilizan para comparar rápidamente las claves del diccionario durante una búsqueda en el diccionario. Los valores numéricos que son iguales tienen el mismo valor hash (incluso si son de diferentes tipos, como es el caso de 1 y 1.0).

1912 help (builtins) — Invoca el sistema de ayuda integrado.

ID 1912 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `help` (builtins)

Explicación española: Invoca el sistema de ayuda integrado.

Sintaxis

`help([objeto])`

Invoca el sistema de ayuda integrado. (Esta función está destinada para uso interactivo). Si no se proporciona ningún argumento, el sistema de ayuda interactivo se inicia en la consola del intérprete. Si el argumento es una cadena, entonces la cadena se busca como el nombre de un módulo, función, clase, método, palabra clave o tema de documentación, y se imprime una página de ayuda en la consola. Si el argumento es cualquier otro tipo de objeto, se genera una página de ayuda sobre el objeto.

Tenga en cuenta que si aparece una barra diagonal (/) en la lista de parámetros de una función, al invocar `help()`, significa que los parámetros anteriores a la barra son solo posicionales. Para más información, consulte la entrada de preguntas frecuentes sobre parámetros solo posicionales.

1913 hex (builtins) — Convierte un número entero en una cadena hexadecimal en minúsculas precedida por "0x".

ID 1913 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `hex` (builtins)

Explicación española: Convierte un número entero en una cadena hexadecimal en minúsculas precedida por "0x".

Sintaxis

`hex(x)`

Convierte un número entero en una cadena hexadecimal en minúsculas precedida por "0x". Si `x` no es un objeto `int` de Python, debe definir un método `index()` que devuelva un entero.

1914 id (builtins) — Devuelve la 'identidad' de un objeto.

ID 1914 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `id` (`builtins`)

Explicación española: Devuelve la 'identidad' de un objeto.

Sintaxis

`id(objeto)`

Devuelve la 'identidad' de un objeto. Este es un entero que se garantiza que es único y constante para este objeto durante su vida útil. Dos objetos con vidas no superpuestas pueden tener el mismo valor de `id()`.

1915 `input` (`builtins`) — Si el argumento mensaje está presente, se escribe en la salida estándar sin un salto de línea final.

ID 1915 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `input` (`builtins`)

Explicación española: Si el argumento mensaje está presente, se escribe en la salida estándar sin un salto de línea final.

Sintaxis

`input([mensaje])`

Si el argumento mensaje está presente, se escribe en la salida estándar sin un salto de línea final. La función luego lee una línea de la entrada, la convierte en una cadena (eliminando un salto de línea final) y la devuelve. Cuando se lee EOF, se genera un `EOFError`. Ejemplo:

1916 `isinstance` (`builtins`) — Devuelve `True` si el argumento `object` es una instancia del argumento `classinfo`, o de una subclase...

ID 1916 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `isinstance` (`builtins`)

Explicación española: Devuelve `True` si el argumento `object` es una instancia del argumento `classinfo`, o de una subclase...

Sintaxis

`isinstance(object, classinfo)`

Devuelve `True` si el argumento `object` es una instancia del argumento `classinfo`, o de una subclase (directa, indirecta o virtual) de este. Si `object` no es un objeto del tipo dado, la función siempre devuelve `False`. Si `classinfo` es una tupla de objetos de tipo (o recursivamente, otras tuplas similares), devuelve `True` si `object` es una instancia de cualquiera de los tipos. Si `classinfo` no es un tipo o una tupla de tipos y tales tuplas, se genera una excepción `TypeError`.

1917 `issubclass` (`builtins`) — Devuelve `True` si la clase es una subclase (directa, indirecta o virtual) de `classinfo`.

ID 1917 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `issubclass` (`builtins`)

Explicación española: Devuelve `True` si la clase es una subclase (directa, indirecta o virtual) de `classinfo`.

Sintaxis

`issubclass(clase, classinfo)`

Devuelve `True` si la clase es una subclase (directa, indirecta o virtual) de `classinfo`. Una clase se considera una subclase de sí misma. `classinfo` puede ser una tupla de objetos de clase, en cuyo caso se verificará cada entrada en `classinfo`. En cualquier otro caso, se genera una excepción `TypeError`.

1918 `iter` (`builtins`) — Devuelve un objeto iterador.

ID 1918 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `iter` (builtins)

Explicación española: Devuelve un objeto iterador.

Sintaxis

`iter(objeto)`

Devuelve un objeto iterador. El argumento `objeto` debe soportar el protocolo de iteración (es decir, implementar el método `__iter__()`) o el protocolo de secuencia (es decir, implementar el método `__getitem__()` con índices enteros que comienzan en 0). Si el objeto no soporta ninguno de estos protocolos, se genera un `TypeError`.

Ejemplo

```
# Usando iter() con objetos personalizados
class Counter:
    def __init__(self, max_count):
        self.max_count = max_count
        self.count = 0
```

```
def __iter__(self):
```

```
    return self
```

```
def __next__(self):
```

```
    if self.count < self.max_count:
```

```
        self.count += 1
```

```
        return self.count
```

```
    raise StopIteration
```

```
counter = Counter(3)
```

```
for num in iter(counter):
```

```
    print(num, end=',') # Salida: 1,2,3,
```

1919 `len` (builtins) — Devuelve la longitud (el número de elementos) de un objeto.

ID 1919 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `len` (builtins)

Explicación española: Devuelve la longitud (el número de elementos) de un objeto.

Sintaxis

`len(s)`

Devuelve la longitud (el número de elementos) de un objeto. El argumento puede ser una secuencia (como una cadena, bytes, una tupla, una lista o un rango) o una colección (como un diccionario, un conjunto o un conjunto congelado).

1920 `locals` (builtins) — Actualiza y devuelve un diccionario que representa la tabla de símbolos local actual.

ID 1920 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `locals` (builtins)

Explicación española: Actualiza y devuelve un diccionario que representa la tabla de símbolos local actual.

Sintaxis

`locals()`

Actualiza y devuelve un diccionario que representa la tabla de símbolos local actual. Las variables libres son devueltas por `locals()` cuando se llama en bloques de función, pero no en bloques de clase. Tenga en cuenta que a nivel de módulo, `locals()` y `globals()` son el mismo diccionario.

1921 `max` (builtins) — Devuelve el elemento más grande en un iterable ``i`` o el más grande de dos o más argumentos.

ID 1921 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `max` (builtins)

Explicación española: Devuelve el elemento más grande en un iterable ``i`` o el más grande de dos o más argumentos.

Sintaxis

```
max(i, *[, key, default])  
max(arg_1, arg_2, *args [, key])
```

Devuelve el elemento más grande en un iterable `i` o el más grande de dos o más argumentos.

Si se proporciona un argumento posicional, debe ser un iterable. Se devuelve el elemento más grande en el iterable. Si se proporcionan dos o más argumentos posicionales, se devuelve el más grande de los argumentos posicionales.

Hay dos argumentos opcionales solo por palabra clave. El argumento `key` especifica una función de orden de un argumento, como la utilizada para `list.sort()`. El argumento `default` especifica un objeto que se devolverá si el iterable proporcionado está vacío. Si el iterable está vacío y no se proporciona `default`, se genera un `ValueError`.

Si varios elementos son máximos, la función devuelve el primero encontrado. Esto es consistente con otras herramientas que preservan la estabilidad del orden, como `sorted(iterable, key=keyfunc, reverse=True)[0]` y `heapq.nlargest(1, iterable, key=keyfunc)`.

1922 min (builtins) — Devuelve el elemento más pequeño en un iterable ``i`` o el más pequeño de dos o más argumentos.

ID 1922 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `min (builtins)`

Explicación española: Devuelve el elemento más pequeño en un iterable ``i`` o el más pequeño de dos o más argumentos.

Sintaxis

```
min(i, *[, key, default])  
min(arg1, arg2, *args [, key])
```

Devuelve el elemento más pequeño en un iterable `i` o el más pequeño de dos o más argumentos.

Si se proporciona un argumento posicional, debe ser un iterable. Se devuelve el elemento más pequeño del iterable. Si se proporcionan dos o más argumentos posicionales, se devuelve el más pequeño de los argumentos posicionales.

Hay dos argumentos opcionales solo por palabra clave. El argumento `key` especifica una función de orden de un argumento, como la utilizada para `list.sort()`. El argumento `default` especifica un objeto que se devolverá si el iterable proporcionado está vacío. Si el iterable está vacío y no se proporciona `default`, se genera un `ValueError`.

Si varios elementos son mínimos, la función devuelve el primero encontrado. Esto es consistente con otras herramientas que preservan la estabilidad del orden, como `sorted(iterable, key=keyfunc)[0]` y `heapq.nsmallest(1, iterable, key=keyfunc)`.

1923 next (builtins) — Devuelve el siguiente elemento del iterador ``iter``.

ID 1923 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `next (builtins)`

Explicación española: Devuelve el siguiente elemento del iterador ``iter``.

Sintaxis

```
next(iter)
```

Devuelve el siguiente elemento del iterador `iter`. Si no hay más elementos, se genera una excepción `StopIteration`.

Ejemplo

```
numbers = [10, 20, 30]  
it = iter(numbers)
```

```
try:  
    while True:  
        print(next(it))  
except StopIteration:  
    print("FIN")
```

1924 oct (builtins) — Convierte un número entero ``x`` en una cadena octal precedida por `"0o"`.

ID 1924 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `oct` (builtins)

Explicación española: Convierte un número entero `x` en una cadena octal precedida por "0o".

Sintaxis

`oct(x)`

Convierte un número entero `x` en una cadena octal precedida por "0o". El resultado es una expresión Python válida. Si `x` no es un objeto `int` de Python, debe definir un método `index()` que devuelva un entero.

1925 `ord` (builtins) — Dada una cadena que representa un solo carácter Unicode, devuelve un entero que representa el punto de...

ID 1925 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `ord` (builtins)

Explicación española: Dada una cadena que representa un solo carácter Unicode, devuelve un entero que representa el punto de...

Sintaxis

`ord(c)`

Dada una cadena que representa un solo carácter Unicode, devuelve un entero que representa el punto de código Unicode de ese carácter. Por ejemplo, `ord('a')` devuelve el entero 97 y `ord('€')` (símbolo del euro) devuelve 8364. Esta es la inversa de `chr()`.

1926 `pow` (builtins) — Esta función calcula la potencia de un número elevando `base` a la potencia de `exp`.

ID 1926 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `pow` (builtins)

Explicación española: Esta función calcula la potencia de un número elevando `base` a la potencia de `exp`.

Sintaxis

`pow(base, exp [, mod])`

Esta función calcula la potencia de un número elevando `base` a la potencia de `exp`. Si se proporciona el parámetro opcional `mod`, calcula `base` elevado a la potencia de `exp`, módulo `mod`, lo que es más eficiente que calcular `pow(base, exp) % mod`. Sin el parámetro `mod`, `pow(base, exp)` funciona de manera idéntica al operador de potencia: `base**exp`.

La función requiere que sus argumentos sean de tipos numéricos. Para tipos de operandos mixtos, se aplican las reglas de coerción para operadores aritméticos binarios. Si `base` y `exp` son de tipo `int`, el resultado también será un entero a menos que `exp` sea negativo. En tales casos, todos los argumentos se convierten a `float`, entregando un resultado de punto flotante. Por ejemplo, `10**2` produce 100, pero `10**-2` produce 0.01.

Cuando tanto `base` como `exp` son enteros y se incluye `mod`, `mod` también debe ser un entero y no cero. Si `exp` es negativo y se proporciona `mod`, `base` debe ser relativamente primo a `mod`. En estas condiciones, el resultado devuelto es `pow(inv_base, -exp, mod)`, donde `inv_base` es la inversa modular de `base` con respecto a `mod`.

1927 `print` (builtins) — Esta función imprime objetos en el flujo de texto especificado (por defecto `sys.stdout`), con cada objeto...

ID 1927 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `print` (builtins)

Explicación española: Esta función imprime objetos en el flujo de texto especificado (por defecto `sys.stdout`), con cada objeto...

Sintaxis

`print(*objects, sep=' ', end='\n', file=sys.stdout)`

Esta función imprime objetos en el flujo de texto especificado (por defecto `sys.stdout`), con cada objeto separado por la cadena `sep`, y seguido por la cadena `end`. Los argumentos `sep`, `end` y `file` deben especificarse como

argumentos de palabra clave si se proporcionan.

Todos los argumentos posicionales se convierten en cadenas utilizando `str()` y se escriben en el flujo, con `sep` entre cada objeto y `end` después del último objeto. Tanto `sep` como `end` deben ser cadenas, pero también pueden ser `None`, lo que indica el uso de valores predeterminados. Si no se proporcionan objetos, `print()` solo mostrará la cadena `end`.

El argumento `file` debe ser un objeto que tenga un método `write(string)`. Si `file` no se proporciona o es `None`, se utiliza `sys.stdout` como el flujo de salida predeterminado. Dado que `print()` convierte todos los argumentos en cadenas de texto, no es adecuado para escribir en objetos de archivo en modo binario. En esos casos, se debe usar `file.write(...)` en su lugar.

1928 repr (builtins) — Devuelve una cadena que contiene una representación imprimible de un objeto.

ID 1928 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>repr (builtins)</code> Explicación española: Devuelve una cadena que contiene una representación imprimible de un objeto.
Sintaxis <code>repr(objeto)</code>

Devuelve una cadena que contiene una representación imprimible de un objeto. Para muchos tipos, esta función intenta devolver una cadena que generaría un objeto con el mismo valor cuando se pasa a `eval()`, de lo contrario, la representación es una cadena encerrada entre signos de menor que y mayor que que contiene el nombre del tipo de objeto junto con información adicional que a menudo incluye el nombre y la dirección del objeto. Una clase puede controlar lo que esta función devuelve para sus instancias definiendo un método `repr()`.

1929 round (builtins) — Redondea `número` a `ndigits` lugares de precisión después del punto decimal.

ID 1929 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>round (builtins)</code> Explicación española: Redondea `número` a `ndigits` lugares de precisión después del punto decimal.
Sintaxis <code>round(número [, ndigits])</code>

Redondea `número` a `ndigits` lugares de precisión después del punto decimal. Si se omite `ndigits` o es `None`, la función devuelve el entero más cercano a la entrada.

Para los tipos que admiten `round()`, los valores se redondean al múltiplo más cercano de 10 elevado a la potencia de menos `ndigits`. Si dos múltiplos son igualmente cercanos, el redondeo se realiza hacia el número par más cercano (por ejemplo, tanto `round(0.5)` como `round(-0.5)` devuelven 0, mientras que `round(1.5)` devuelve 2). `ndigits` puede ser cualquier valor entero, ya sea positivo, cero o negativo.

El tipo de retorno de la función coincide con el tipo de `número` si se especifica `ndigits`. Si se omite `ndigits` o es `None`, el tipo de retorno es un entero.

1930 sorted (builtins) — Devuelve una nueva lista que contiene todos los elementos del `iterable`, ordenados en orden ascendente...

ID 1930 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>sorted (builtins)</code> Explicación española: Devuelve una nueva lista que contiene todos los elementos del `iterable`, ordenados en orden ascendente...
Sintaxis <code>sorted(iterable, *, key=None, reverse=False)</code>

Devuelve una nueva lista que contiene todos los elementos del `iterable`, ordenados en orden ascendente por defecto.

Hay dos argumentos de palabra clave opcionales. El argumento `key` especifica una función de un argumento que se utiliza para extraer una clave de comparación de cada elemento del `iterable`. El valor predeterminado de `key` es `None`, lo que significa que los elementos se comparan directamente. El argumento `reverse` es un booleano; cuando

se establece en True, los elementos se ordenan como si cada comparación se invirtiera, lo que resulta en un orden descendente.

Ejemplo

```
pairs = [(1, 'B'), (3, 'A'), (2, 'C')]
# Ordenando por el segundo elemento de la tupla
sorted_pairs = sorted(pairs, key=lambda x: x[1])
print(sorted_pairs) # [(3, 'A'), (1, 'B'), (2, 'C')]
```

1931 sum (builtins) — Calcula el total sumando los elementos del objeto iterable especificado (`iterable`) al valor inicial...

ID 1931 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `sum (builtins)`

Explicación española: Calcula el total sumando los elementos del objeto iterable especificado (`iterable`) al valor inicial...

Sintaxis

`sum(iterable, /, start=0)`

Calcula el total sumando los elementos del objeto iterable especificado (`iterable`) al valor inicial `start`, sumando de izquierda a derecha. Los elementos en el `iterable` deben ser números, y el valor `start` no puede ser una cadena.

Aunque `sum()` se usa comúnmente para este propósito, hay alternativas especializadas para escenarios específicos:

- Para concatenar cadenas, use `' '.join(sequence)`.
- Para sumar números de punto flotante con mayor precisión, se recomienda `math.fsum()`.
- Para concatenar múltiples objetos iterables, `itertools.chain()` ofrece un método adecuado.

Esta explicación asegura que el término `iterable` se entienda como un objeto que se puede iterar, alineándose con su definición técnica.

1932 open (builtins) — La función `open()` abre un archivo y devuelve el objeto de archivo de tipo `TextIOWrapper`.

ID 1932 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `open (builtins)`

Explicación española: La función `open()` abre un archivo y devuelve el objeto de archivo de tipo `TextIOWrapper`.

Sintaxis

`open(filename, mode='r', buffering=-1, encoding='utf-8')`

La función `open()` abre un archivo y devuelve el objeto de archivo de tipo `TextIOWrapper`. Si no se puede abrir el archivo, se genera un `OSError`. En esta implementación de MicroPython, esta función está limitada al procesamiento de archivos de texto, por lo tanto, si necesita trabajar con archivos binarios, se recomienda usar la clase `FileIO` del módulo `uio`.

Parámetros

- `filename`: Un objeto similar a una ruta que representa la ruta del archivo (absoluta o relativa al directorio de trabajo actual) que se va a abrir.
- `mode`: Una cadena opcional que especifica el modo en el que se abre el archivo. El valor predeterminado es `'r'`, lo que significa abrir para lectura en modo texto. Otros modos disponibles son:

`'r'`: Abrir para lectura (predeterminado).

`'w'`: Abrir para escritura, truncando primero el archivo.

`'a'`: El archivo se abre para escritura con el puntero situado al final del archivo. Los nuevos datos se añaden al final del archivo.

Nota: Esta implementación de MicroPython no distingue entre modos binarios y de texto. Modos como `'rb'`, `'wb'` y `'ab'` se tratan de la misma manera que sus contrapartes de texto, `'rt'`, `'wt'` y `'at'` (o simplemente `'r'`, `'w'` y `'a'`). Los archivos siempre se tratan como archivos de texto (tipo `TextIOWrapper`).

- `buffering`: Un entero opcional que se utiliza para controlar la política de almacenamiento en búfer. `-1` (predeterminado) significa usar la configuración predeterminada del sistema, `0` significa sin almacenamiento en búfer, y cualquier entero positivo especifica el tamaño del búfer.

- **encoding**: Un parámetro opcional que especifica la codificación utilizada para decodificar o codificar el archivo. La codificación predeterminada y única admitida es UTF-8.

Consulte la descripción del objeto `TextIOWrapper` para obtener detalles sobre los métodos disponibles para las operaciones de archivo.

2127 Módulo `cas` de Python

ID 2127 · título recuperado de la página española

El módulo `cas` proporciona algunos métodos para pasar expresiones al CAS para su evaluación.

2128 `caseval (cas)` — Acepta una expresión como una cadena.

ID 2128 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `caseval (cas)`

Explicación española: Acepta una expresión como una cadena.

Sintaxis

`caseval(expresión)`

Acepta una expresión como una cadena. Devuelve la expresión evaluada por el CAS.

Sinónimos: `eval_expr`, `xcas`

2129 `get_key (cas)` — Espera la próxima pulsación de tecla y devuelve un valor entero de 0 a 50, que identifica teclas...

ID 2129 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `get_key (cas)`

Explicación española: Espera la próxima pulsación de tecla y devuelve un valor entero de 0 a 50, que identifica teclas...

Sintaxis

`get_key()`

Espera la próxima pulsación de tecla y devuelve un valor entero de 0 a 50, que identifica teclas individuales en el teclado dispuestas en filas de arriba hacia abajo (comenzando por 'Apps' y terminando en '+').

2171 Módulo `cmath` de Python

ID 2171 · título recuperado de la página española

El módulo `cmath` proporciona algunas funciones matemáticas básicas para trabajar con números complejos.

2172 `e (cmath)` — Devuelve el valor de la constante e utilizada por el módulo `cmath` (2.71828182845905).

ID 2172 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `e (cmath)`

Explicación española: Devuelve el valor de la constante e utilizada por el módulo `cmath` (2.71828182845905).

Sintaxis

`e`

Devuelve el valor de la constante e utilizada por el módulo `cmath` (2.71828182845905).

2173 `pi (cmath)` — Devuelve el valor de la constante π utilizada por el módulo `cmath` (3.14159265358979).

ID 2173 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `pi (cmath)`

Explicación española: Devuelve el valor de la constante π utilizada por el módulo `cmath` (3.14159265358979).

Sintaxis

`pi`

Devuelve el valor de la constante π utilizada por el módulo `cmath` (3.14159265358979).

2174 phase (cmath) — Devuelve el ángulo del número complejo `z` en radianes, conocido como la fase.

ID 2174 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `phase (cmath)`

Explicación española: Devuelve el ángulo del número complejo `z` en radianes, conocido como la fase.

Sintaxis

`phase(z)`

Devuelve el ángulo del número complejo `z` en radianes, conocido como la fase. El resultado es el ángulo desde el eje real positivo hasta el vector que representa `z`, con valores en el rango $-\pi < \text{ángulo} \leq \pi$.

2175 polar (cmath) — Devuelve, como una tupla, la forma polar de `z`.

ID 2175 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `polar (cmath)`

Explicación española: Devuelve, como una tupla, la forma polar de `z`.

Sintaxis

`polar(z)`

Devuelve, como una tupla, la forma polar de `z`.

2176 rect (cmath) — Devuelve el número complejo correspondiente a las coordenadas polares con módulo `r` y fase `phi`.

ID 2176 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `rect (cmath)`

Explicación española: Devuelve el número complejo correspondiente a las coordenadas polares con módulo `r` y fase `phi`.

Sintaxis

`rect(r, phi)`

Devuelve el número complejo correspondiente a las coordenadas polares con módulo `r` y fase `phi`.

2177 exp (cmath) — Devuelve la exponencial de un número complejo `z`, equivalente a e elevado a la potencia de `z`.

ID 2177 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `exp (cmath)`

Explicación española: Devuelve la exponencial de un número complejo `z`, equivalente a e elevado a la potencia de `z`.

Sintaxis

`exp(z)`

Devuelve la exponencial de un número complejo `z`, equivalente a e elevado a la potencia de `z`.

Ejemplo

```
import cmath
z = cmath.exp(cmath.pi * 1j) # Usando la fórmula de Euler
print(z) # Salida: (-1+1.224646799147353e-16j)
```

2178 log (cmath) — Devuelve el logaritmo natural de un número complejo `z` para la base `e`.

ID 2178 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `log (cmath)`

Explicación española: Devuelve el logaritmo natural de un número complejo `z` para la base 'e'.

Sintaxis

`log(z)`

Devuelve el logaritmo natural de un número complejo `z` para la base 'e'.

2179 log10 (cmath) — Devuelve el logaritmo en base 10 de un número complejo `z`.

ID 2179 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `log10 (cmath)`

Explicación española: Devuelve el logaritmo en base 10 de un número complejo `z`.

Sintaxis

`log10(z)`

Devuelve el logaritmo en base 10 de un número complejo `z`.

2180 sqrt (cmath) — Devuelve la raíz cuadrada de un número complejo `z`.

ID 2180 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `sqrt (cmath)`

Explicación española: Devuelve la raíz cuadrada de un número complejo `z`.

Sintaxis

`sqrt(z)`

Devuelve la raíz cuadrada de un número complejo `z`.

2182 cos (cmath) — Devuelve el coseno de un número complejo `z`.

ID 2182 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `cos (cmath)`

Explicación española: Devuelve el coseno de un número complejo `z`.

Sintaxis

`cos(z)`

Devuelve el coseno de un número complejo `z`.

2181 sin (cmath) — Devuelve el seno de un número complejo `z`.

ID 2181 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `sin (cmath)`

Explicación española: Devuelve el seno de un número complejo `z`.

Sintaxis

`sin(z)`

Devuelve el seno de un número complejo `z`.

2199 Módulo gc de Python

ID 2199 · título recuperado de la página española

El módulo `gc` proporciona una interfaz para el recolector de basura opcional. Permite la activación y desactivación del recolector, así como la verificación de su estado. También permite la iniciación manual del proceso de

recolección de basura (a demanda). Además, hay funciones disponibles para monitorear la memoria libre y asignada, y para gestionar el umbral en el que se activa la recolección automática de basura, permitiendo ajustar la frecuencia de recolección para satisfacer los requisitos específicos de la aplicación.

2200 collect (gc) — Inicia un ciclo de recolección de basura.

ID 2200 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: collect (gc)

Explicación española: Inicia un ciclo de recolección de basura.

Sintaxis

collect()

Inicia un ciclo de recolección de basura. Esta función fuerza al recolector de basura a buscar objetos que ya no son accesibles en el programa y libera la memoria que ocupan. Ayuda a gestionar la memoria activando manualmente la limpieza de la memoria, lo cual es especialmente útil después de completar operaciones intensivas en memoria.

Aunque collect() puede ayudar a recuperar memoria, su invocación manual debe usarse con moderación ya que es computacionalmente costosa. El uso frecuente puede llevar a una degradación del rendimiento, particularmente en sistemas con restricciones estrictas de tiempo de ejecución.

Es aconsejable confiar en la recolección automática de basura para la gestión rutinaria de la memoria. La intervención manual a través de collect() debe considerarse en escenarios donde la recuperación inmediata de memoria es crítica y supera la posible ralentización durante su ejecución.

2201 disable (gc) — Desactiva la recolección automática de basura.

ID 2201 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: disable (gc)

Explicación española: Desactiva la recolección automática de basura.

Sintaxis

disable()

Desactiva la recolección automática de basura. La memoria del montón aún se puede asignar, y la recolección de basura aún se puede iniciar manualmente usando gc.collect().

2202 enable (gc) — Reactiva la recolección automática de basura si ha sido deshabilitada previamente usando el método...

ID 2202 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: enable (gc)

Explicación española: Reactiva la recolección automática de basura si ha sido deshabilitada previamente usando el método...

Sintaxis

enable()

Reactiva la recolección automática de basura si ha sido deshabilitada previamente usando el método disable(). Se utiliza típicamente para restaurar la gestión automática de memoria después de deshabilitarla temporalmente por razones de rendimiento durante operaciones críticas.

2203 isenabled (gc) — Devuelve el estado actual de la recolección automática de basura.

ID 2203 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: isenabled (gc)

Explicación española: Devuelve el estado actual de la recolección automática de basura.

Sintaxis

isenabled()

Devuelve el estado actual de la recolección automática de basura. Indica si el recolector de basura está habilitado (True) o deshabilitado (False).

2204 mem_free (gc) — Devuelve el número de bytes de RAM de montón disponibles, o -1 si esta cantidad no se conoce.

ID 2204 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: mem_free (gc) Explicación española: Devuelve el número de bytes de RAM de montón disponibles, o -1 si esta cantidad no se conoce.
Sintaxis mem_free()

Devuelve el número de bytes de RAM de montón disponibles, o -1 si esta cantidad no se conoce.

2205 mem_alloc (gc) — Devuelve el número de bytes de RAM de montón que están asignados.

ID 2205 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: mem_alloc (gc) Explicación española: Devuelve el número de bytes de RAM de montón que están asignados.
Sintaxis mem_alloc()

Devuelve el número de bytes de RAM de montón que están asignados.

2206 threshold (gc) — Establecer o consultar el umbral de asignación adicional de GC.

ID 2206 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: threshold (gc) Explicación española: Establecer o consultar el umbral de asignación adicional de GC.
Sintaxis threshold([cantidad])

Establecer o consultar el umbral de asignación adicional de GC. Normalmente, una recolección se desencadena solo cuando una nueva asignación no puede ser satisfecha, es decir, en una condición de falta de memoria (OOM). Si se llama a esta función, además de OOM, se desencadenará una recolección cada vez que se hayan asignado cantidad de bytes (en total, desde la última vez que se asignó dicha cantidad de bytes). La cantidad generalmente se especifica como menor que el tamaño total del montón, con la intención de desencadenar una recolección antes de que el montón se agote, y con la esperanza de que una recolección temprana prevenga una fragmentación excesiva de la memoria. Esta es una medida heurística, cuyo efecto variará de una aplicación a otra, así como el valor óptimo del parámetro cantidad.

Llamar a la función sin argumento devolverá el valor actual del umbral. Un valor de -1 significa un umbral de asignación deshabilitado.

2047 Módulo graphic de Python

ID 2047 · título recuperado de la página española

El módulo graphic incluye un conjunto de comandos gráficos basados en píxeles que permiten crear dibujos.

2048 cyan (graphic) — La constante `cyan` tiene un valor de 65535, correspondiente al color cian en el modelo RGB888 de 24...

ID 2048 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: cyan (graphic) Explicación española: La constante `cyan` tiene un valor de 65535, correspondiente al color cian en el modelo RGB888 de 24...

La constante cyan tiene un valor de 65535, correspondiente al color cian en el modelo RGB888 de 24 bits, representado como (0, 255, 255).

2049 magenta (graphic) — La constante `magenta` tiene un valor de 16711935, correspondiente al color magenta en el modelo RGB888...

ID 2049 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: magenta (graphic)

Explicación española: La constante `magenta` tiene un valor de 16711935, correspondiente al color magenta en el modelo RGB888...

La constante magenta tiene un valor de 16711935, correspondiente al color magenta en el modelo RGB888 de 24 bits, representado como (255, 0, 255).

2050 yellow (graphic) — La constante `yellow` tiene un valor de 16776960, correspondiente al color amarillo en el modelo RGB888...

ID 2050 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: yellow (graphic)

Explicación española: La constante `yellow` tiene un valor de 16776960, correspondiente al color amarillo en el modelo RGB888...

La constante yellow tiene un valor de 16776960, correspondiente al color amarillo en el modelo RGB888 de 24 bits, representado como (255, 255, 0).

2051 blue (graphic) — La constante `blue` tiene un valor de 255, correspondiente al color azul en el modelo RGB888 de 24 bits,...

ID 2051 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: blue (graphic)

Explicación española: La constante `blue` tiene un valor de 255, correspondiente al color azul en el modelo RGB888 de 24 bits,...

La constante blue tiene un valor de 255, correspondiente al color azul en el modelo RGB888 de 24 bits, representado como (0, 0, 255).

2052 red (graphic) — La constante `red` tiene un valor de 16711680, correspondiente al color rojo en el modelo RGB888 de 24...

ID 2052 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: red (graphic)

Explicación española: La constante `red` tiene un valor de 16711680, correspondiente al color rojo en el modelo RGB888 de 24...

La constante red tiene un valor de 16711680, correspondiente al color rojo en el modelo RGB888 de 24 bits, representado como (255, 0, 0).

2053 green (graphic) — La constante `green` tiene un valor de 65280, correspondiente al color verde en el modelo RGB888 de 24...

ID 2053 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: green (graphic)

Explicación española: La constante `green` tiene un valor de 65280, correspondiente al color verde en el modelo RGB888 de 24...

La constante green tiene un valor de 65280, correspondiente al color verde en el modelo RGB888 de 24 bits, representado como (0, 255, 0).

2054 black (graphic) — La constante `black` tiene un valor de 0, correspondiente al color negro en el modelo RGB888 de 24 bits,...

ID 2054 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: black (graphic)

Explicación española: La constante `black` tiene un valor de 0, correspondiente al color negro en el modelo RGB888 de 24 bits,...

La constante black tiene un valor de 0, correspondiente al color negro en el modelo RGB888 de 24 bits, representado como (0, 0, 0).

2055 white (graphic) — La constante `white` tiene un valor de 16777215, correspondiente al color blanco en el modelo RGB888 de...

ID 2055 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: white (graphic)

Explicación española: La constante `white` tiene un valor de 16777215, correspondiente al color blanco en el modelo RGB888 de...

La constante white tiene un valor de 16777215, correspondiente al color blanco en el modelo RGB888 de 24 bits, representado como (255, 255, 255).

2056 draw_pixel (graphic) — Establece el color del píxel en las coordenadas (x, y) en la pantalla.

ID 2056 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: draw_pixel (graphic)

Explicación española: Establece el color del píxel en las coordenadas (x, y) en la pantalla.

Sintaxis

draw_pixel(x, y [, color=0])

Establece el color del píxel en las coordenadas (x, y) en la pantalla.

Sinónimos: set_pixel

2057 draw_line (graphic) — Dibuja una línea entre los puntos (x1, y1) y (x2, y2) en la pantalla utilizando el color especificado.

ID 2057 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: draw_line (graphic)

Explicación española: Dibuja una línea entre los puntos (x1, y1) y (x2, y2) en la pantalla utilizando el color especificado.

Sintaxis

draw_line(x1, y1, x2, y2 [, color=0])

Dibuja una línea entre los puntos (x1, y1) y (x2, y2) en la pantalla utilizando el color especificado.

2058 draw_polygon (graphic) — Dibuja un polígono definido por una lista o tupla de coordenadas (`puntos`) utilizando el color...

ID 2058 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: draw_polygon (graphic)

Explicación española: Dibuja un polígono definido por una lista o tupla de coordenadas (`puntos`) utilizando el color...

Sintaxis

draw_polygon(puntos [, color=0])

Dibuja un polígono definido por una lista o tupla de coordenadas (puntos) utilizando el color especificado. El polígono se cierra automáticamente conectando el último punto con el primero; por lo tanto, no es necesario repetir la primera coordenada. El polígono no está relleno; solo se dibuja el perímetro.

Ejemplo

```
draw_polygon([(50, 50), (150, 50), (120, 150), (80, 150)], green)
```

2059 draw_filled_polygon (graphic) — Dibuja un polígono relleno definido por una lista o tupla de coordenadas ('puntos') utilizando el color...

ID 2059 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: draw_filled_polygon (graphic)

Explicación española: Dibuja un polígono relleno definido por una lista o tupla de coordenadas ('puntos') utilizando el color...

Sintaxis

```
draw_filled_polygon(puntos [, color=0])
```

Dibuja un polígono relleno definido por una lista o tupla de coordenadas (puntos) utilizando el color especificado. El polígono se cierra automáticamente conectando el último punto con el primero; por lo tanto, no es necesario repetir la primera coordenada.

Ejemplo

```
draw_filled_polygon([(100, 50), (180, 100), (100, 150), (20, 100)], blue)
```

2060 draw_filled_rectangle (graphic) — Dibuja un rectángulo relleno en la pantalla, definido por su esquina superior izquierda en ('x', 'y'),...

ID 2060 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: draw_filled_rectangle (graphic)

Explicación española: Dibuja un rectángulo relleno en la pantalla, definido por su esquina superior izquierda en ('x', 'y'),...

Sintaxis

```
draw_filled_rectangle(x, y, ancho, alto [, color=0])
```

Dibuja un rectángulo relleno en la pantalla, definido por su esquina superior izquierda en (x, y), su ancho y su alto, utilizando el color especificado.

Sinónimos: fill_rect, draw_rectangle

2061 draw_circle (graphic) — Dibuja un círculo en la pantalla, definido por su centro en ('x', 'y') y su 'radio', utilizando el...

ID 2061 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: draw_circle (graphic)

Explicación española: Dibuja un círculo en la pantalla, definido por su centro en ('x', 'y') y su 'radio', utilizando el...

Sintaxis

```
draw_circle(x, y, radio [, color=0])
```

Dibuja un círculo en la pantalla, definido por su centro en (x, y) y su radio, utilizando el color especificado.

2062 draw_filled_circle (graphic) — Dibuja un círculo relleno en la pantalla, definido por su centro en ('x', 'y') y su 'radio', utilizando...

ID 2062 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `draw_filled_circle` (graphic)

Explicación española: Dibuja un círculo relleno en la pantalla, definido por su centro en (`x`, `y`) y su `radio`, utilizando...

Sintaxis

```
draw_filled_circle(x, y, radio [, color=0])
```

Dibuja un círculo relleno en la pantalla, definido por su centro en (`x`, `y`) y su `radio`, utilizando el `color` especificado.

2063 `draw_arc` (graphic) — Dibuja un arco cerrado en la pantalla, definido por un segmento elíptico con centro en (`x`, `y`).

ID 2063 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `draw_arc` (graphic)

Explicación española: Dibuja un arco cerrado en la pantalla, definido por un segmento elíptico con centro en (`x`, `y`).

Sintaxis

```
draw_arc(x, y, radio_x, radio_y, ángulo_inicio, ángulo_fin [, color])
```

Dibuja un arco cerrado en la pantalla, definido por un segmento elíptico con centro en (`x`, `y`). El arco se dibuja entre los ángulos `ángulo_inicio` y `ángulo_fin`, con radios especificados a lo largo de los ejes `x` e `y`, y utilizando el `color` especificado. Esta función crea una forma cerrada al conectar los puntos finales del arco con el centro, formando un segmento similar a un pastel.

2064 `draw_filled_arc` (graphic) — Dibuja un sector relleno de una elipse en la pantalla, definido por un segmento de arco con centro en...

ID 2064 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `draw_filled_arc` (graphic)

Explicación española: Dibuja un sector relleno de una elipse en la pantalla, definido por un segmento de arco con centro en...

Sintaxis

```
draw_filled_arc(x, y, x_radius, y_radius, angle_start, angle_end [, color=0])
```

Dibuja un sector relleno de una elipse en la pantalla, definido por un segmento de arco con centro en (`x`, `y`). El sector se dibuja entre los ángulos `angle_start` y `angle_end`, con radios especificados a lo largo de los ejes `x` e `y`. Esta función crea una forma cerrada, similar a un pastel, y la llena con el `color` especificado.

2065 `get_pixel` (graphic) — Recupera el color del píxel ubicado en las coordenadas especificadas (`x`, `y`).

ID 2065 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `get_pixel` (graphic)

Explicación española: Recupera el color del píxel ubicado en las coordenadas especificadas (`x`, `y`).

Sintaxis

```
get_pixel(x, y [, format])
```

Recupera el color del píxel ubicado en las coordenadas especificadas (`x`, `y`). El parámetro opcional `format` determina el formato en el que se devuelve el color. Cuando el formato se establece en 565, la función devuelve el color como un entero de 16 bits en el formato RGB565. Cuando el formato se establece en 888, la función devuelve el color como un valor entero de 24 bits en el formato RGB888. Si el formato se establece en cualquier otro valor o se omite, el color se devuelve como una tupla (`r`, `g`, `b`) en el formato RGB565.

2066 `draw_string` (graphic) — Dibuja un texto en la pantalla en las coordenadas especificadas (`x`, `y`).

ID 2066 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `draw_string` (graphic)

Explicación española: Dibuja un texto en la pantalla en las coordenadas especificadas (x, y).

Sintaxis

```
draw_string(x, y, texto [, color_texto [, color_fondo]])
```

Dibuja un texto en la pantalla en las coordenadas especificadas (x, y). El texto se muestra utilizando el color de texto y el color de fondo especificados. Si no se proporcionan colores, el texto se dibuja en negro sobre un fondo blanco.

2067 show_screen (graphic) — Congela la pantalla y espera a que se presione cualquier tecla antes de continuar.

ID 2067 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `show_screen` (graphic)

Explicación española: Congela la pantalla y espera a que se presione cualquier tecla antes de continuar.

Sintaxis

```
show_screen()
```

Congela la pantalla y espera a que se presione cualquier tecla antes de continuar. Esta función es particularmente útil para evitar la sobrescritura inmediata de la salida gráfica después de que un programa finaliza su ejecución o después de ejecutar un comando desde la línea de comandos.

Sinónimos: `show`

Ejemplo

```
import graphic as g; g.draw_rectangle(10,10,50,50,g.red); g.show_screen()
```

2068 clear_screen (graphic) — Limpia la pantalla llenándola con el `color` especificado.

ID 2068 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `clear_screen` (graphic)

Explicación española: Limpia la pantalla llenándola con el `color` especificado.

Sintaxis

```
clear_screen([color])
```

Limpia la pantalla llenándola con el color especificado. Si no se especifica ningún color, la pantalla se llena de blanco.

Sinónimos: `clear`

2253 Módulo Python hprime

ID 2253 · título recuperado de la página española

El módulo `hprime` es la interfaz principal entre Python y el subsistema HP Prime.

Proporciona principalmente primitivas de dibujo para gráficos rápidos y la función `eval` que permite a Python interactuar con el resto del sistema Prime.

Todas las primitivas de dibujo son muy similares a la equivalente de PPL con un par de diferencias:

- La gráfica de destino (G0-G9) debe especificarse siempre. Al igual que en PPL, la gráfica 0 es la pantalla, mientras que las demás son objetos gráficos que se pueden usar libremente (después de redimensionarlos).
- Las funciones que operan en coordenadas cartesianas tienen el sufijo `'_c'`.
- Las funciones tienen menos argumentos opcionales.

2254 pixon (hprime) — Establece el color de un solo píxel en (`x`, `y`) en el gráfico especificado.

ID 2254 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `pixon` (hprime)

Explicación española: Establece el color de un solo píxel en (`x`, `y`) en el gráfico especificado.

Sintaxis

`pixon(gráfico, x, y, color)`

Establece el color de un solo píxel en (`x`, `y`) en el gráfico especificado.

2270 `pixon_c` (hprime) — Establece el color de un solo píxel en las coordenadas cartesianas (`x`, `y`) en el gráfico especificado.

ID 2270 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `pixon_c` (hprime)

Explicación española: Establece el color de un solo píxel en las coordenadas cartesianas (`x`, `y`) en el gráfico especificado.

Sintaxis

`pixon_c(gráfico, x, y, color)`

Establece el color de un solo píxel en las coordenadas cartesianas (`x`, `y`) en el gráfico especificado.

2255 `circle` (hprime) — Dibuja un círculo en el `gráfico` especificado, centrado en (`x`, `y`), con un `radio` y un `color`...

ID 2255 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `circle` (hprime)

Explicación española: Dibuja un círculo en el `gráfico` especificado, centrado en (`x`, `y`), con un `radio` y un `color`...

Sintaxis

`circle(gráfico, x, y, radio, color)`

Dibuja un círculo en el gráfico especificado, centrado en (`x`, `y`), con un radio y un color dados. El radio se mide en píxeles.

2271 `circle_c` (hprime) — Dibuja un círculo en el `gráfico` especificado, centrado en las coordenadas cartesianas (`x`, `y`), con...

ID 2271 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `circle_c` (hprime)

Explicación española: Dibuja un círculo en el `gráfico` especificado, centrado en las coordenadas cartesianas (`x`, `y`), con...

Sintaxis

`circle_c(gráfico, x, y, radio, color)`

Dibuja un círculo en el gráfico especificado, centrado en las coordenadas cartesianas (`x`, `y`), con un radio y un color dados. El radio se mide en unidades cartesianas.

2256 `arc` (hprime) — Dibuja un arco de un círculo en el `gráfico` especificado, centrado en (`x`, `y`), con un `radio` y un...

ID 2256 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `arc` (hprime)

Explicación española: Dibuja un arco de un círculo en el `gráfico` especificado, centrado en (`x`, `y`), con un `radio` y un...

Sintaxis

`arc(gráfico, x, y, radio, color, ángulo_inicio, ángulo_fin)`

Dibuja un arco de un círculo en el gráfico especificado, centrado en (x, y), con un radio y un color dados. El arco comienza en ángulo_inicio y termina en ángulo_fin, con ángulos expresados en grados. El radio se mide en píxeles.

2272 arc_c (hprime) — Dibuja un arco de un círculo en el `gráfico` especificado, centrado en las coordenadas cartesianas (`x`,...

ID 2272 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: arc_c (hprime)

Explicación española: Dibuja un arco de un círculo en el `gráfico` especificado, centrado en las coordenadas cartesianas (`x`,...

Sintaxis

arc_c(gráfico, x, y, radio, color, ángulo_inicio, ángulo_fin)

Dibuja un arco de un círculo en el gráfico especificado, centrado en las coordenadas cartesianas (x, y), con un radio y un color dados. El arco comienza en ángulo_inicio y termina en ángulo_fin, con ángulos expresados en grados. El radio se mide en unidades cartesianas.

2259 blit (hprime) — Copia todo el `gráfico\_fuente` en el `gráfico\_objetivo` en las coordenadas especificadas (`x`, `y`).

ID 2259 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: blit (hprime)

Explicación española: Copia todo el `gráfico\_fuente` en el `gráfico\_objetivo` en las coordenadas especificadas (`x`, `y`).

Sintaxis

blit(gráfico_objetivo, x, y, gráfico_fuente)

Copia todo el gráfico_fuente en el gráfico_objetivo en las coordenadas especificadas (x, y).

2275 blit_c (hprime) — Copia todo el `gráfico\_fuente` en el `gráfico\_objetivo` en las coordenadas cartesianas especificadas...

ID 2275 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: blit_c (hprime)

Explicación española: Copia todo el `gráfico\_fuente` en el `gráfico\_objetivo` en las coordenadas cartesianas especificadas...

Sintaxis

blit_c(gráfico_objetivo, x, y, gráfico_fuente)

Copia todo el gráfico_fuente en el gráfico_objetivo en las coordenadas cartesianas especificadas (x, y).

2257 invert (hprime) — Invierte un rectángulo en el `gráfico` especificado en (`x`, `y`) con las dimensiones especificadas...

ID 2257 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: invert (hprime)

Explicación española: Invierte un rectángulo en el `gráfico` especificado en (`x`, `y`) con las dimensiones especificadas...

Sintaxis

invert(gráfico, x, y, ancho, alto)

Invierte un rectángulo en el gráfico especificado en (x, y) con las dimensiones especificadas ancho y alto, produciendo un efecto de video inverso.

2273 invert_c (hprime) — Invierte un rectángulo en el `gráfico` especificado en las coordenadas cartesianas (`x`, `y`) con las...

Referencia traducidaIdentificador técnico: `invert_c` (hprime)

Explicación española: Invierte un rectángulo en el `gráfico` especificado en las coordenadas cartesianas (`x`, `y`) con las...

Sintaxis`invert_c(gráfico, x, y, ancho, alto)`

Invierte un rectángulo en el gráfico especificado en las coordenadas cartesianas (x, y) con las dimensiones especificadas ancho y alto medidas en unidades cartesianas, produciendo un efecto de video inverso.

2258 fillpoly (hprime) — Dibuja un polígono relleno en el gráfico especificado por `graphic` utilizando coordenadas de píxeles.

Referencia traducidaIdentificador técnico: `fillpoly` (hprime)

Explicación española: Dibuja un polígono relleno en el gráfico especificado por `graphic` utilizando coordenadas de píxeles.

Sintaxis`fillpoly(graphic, vertices, color)`

Dibuja un polígono relleno en el gráfico especificado por `graphic` utilizando coordenadas de píxeles.

El parámetro `vertices` es una lista de tuplas (x, y) que definen los vértices del polígono. El polígono se rellena con el color especificado por `color`.

2274 fillpoly_c (hprime) — Dibuja un polígono relleno en el gráfico especificado por `graphic` utilizando coordenadas cartesianas.

Referencia traducidaIdentificador técnico: `fillpoly_c` (hprime)

Explicación española: Dibuja un polígono relleno en el gráfico especificado por `graphic` utilizando coordenadas cartesianas.

Sintaxis`fillpoly_c(graphic, vertices, color)`

Dibuja un polígono relleno en el gráfico especificado por `graphic` utilizando coordenadas cartesianas.

El parámetro `vertices` es una lista de tuplas (x, y) que definen los vértices del polígono. El polígono se rellena con el color especificado por `color`.

2260 strblit (hprime) — Dibuja el gráfico_fuente en el gráfico_objetivo en (x, y), redimensionándolo a la anchura y altura...

Referencia traducidaIdentificador técnico: `strblit` (hprime)

Explicación española: Dibuja el gráfico_fuente en el gráfico_objetivo en (x, y), redimensionándolo a la anchura y altura...

Sintaxis`strblit(gráfico_objetivo, x, y, ancho, alto, gráfico_fuente)`

Dibuja el gráfico_fuente en el gráfico_objetivo en (x, y), redimensionándolo a la anchura y altura especificadas.

2276 strblit_c (hprime) — Dibuja `gráfico\_fuente` en `gráfico\_objetivo` en las coordenadas cartesianas (`x`, `y`),...

Referencia traducida

Identificador técnico: `strblit_c` (hprime)

Explicación española: Dibuja ``gráfico_fuente`` en ``gráfico_objetivo`` en las coordenadas cartesianas (``x``, ``y``),...

Sintaxis

`strblit_c(gráfico_objetivo, x, y, ancho, alto, gráfico_fuente)`

Dibuja `gráfico_fuente` en `gráfico_objetivo` en las coordenadas cartesianas (`x`, `y`), redimensionándolo a la `ancho` y `alto` especificados medidos en unidades cartesianas.

2261 strblit2 (hprime) — Dibuja un área específica de ``gráfico_fuente`` en las coordenadas (``fuente_x``, ``fuente_y``) con...

ID 2261 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `strblit2` (hprime)

Explicación española: Dibuja un área específica de ``gráfico_fuente`` en las coordenadas (``fuente_x``, ``fuente_y``) con...

Sintaxis

`strblit2(gráfico_objetivo, objetivo_x, objetivo_y, objetivo_ancho, objetivo_alto, gráfico_fuente, fuente_x, fuente_y, fuente_ancho, fuente_alto)`

Dibuja un área específica de `gráfico_fuente` en las coordenadas (`fuente_x`, `fuente_y`) con dimensiones `fuente_ancho` y `fuente_alto`, sobre `gráfico_objetivo` en (`objetivo_x`, `objetivo_y`). El área se redimensiona para ajustarse a las dimensiones `objetivo_ancho` y `objetivo_alto`.

2277 strblit2_c (hprime) — Dibuja un área específica de ``gráfico_fuente`` en (``fuente_x``, ``fuente_y``) con dimensiones ``ancho_fuente``...

ID 2277 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `strblit2_c` (hprime)

Explicación española: Dibuja un área específica de ``gráfico_fuente`` en (``fuente_x``, ``fuente_y``) con dimensiones ``ancho_fuente``...

Sintaxis

`strblit2_c(gráfico_objetivo, objetivo_x, objetivo_y, ancho_objetivo, alto_objetivo, gráfico_fuente, fuente_x, fuente_y, ancho_fuente, alto_fuente)`

Dibuja un área específica de `gráfico_fuente` en (`fuente_x`, `fuente_y`) con dimensiones `ancho_fuente` y `alto_fuente`, sobre `gráfico_objetivo` en (`objetivo_x`, `objetivo_y`) utilizando coordenadas cartesianas. El área se redimensiona para ajustarse a las dimensiones `ancho_objetivo` y `alto_objetivo`. El ancho y la altura se miden en unidades cartesianas.

2262 dimgrob (hprime) — Redimensiona ``gráfico`` a las dimensiones especificadas ``ancho`` y ``alto``, y lo llena con el ``color``...

ID 2262 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `dimgrob` (hprime)

Explicación española: Redimensiona ``gráfico`` a las dimensiones especificadas ``ancho`` y ``alto``, y lo llena con el ``color``...

Sintaxis

`dimgrob(gráfico, ancho, alto, color)`

Redimensiona `gráfico` a las dimensiones especificadas `ancho` y `alto`, y lo llena con el color especificado. El color debe ser un número entero de 32 bits en formato AARRGGBB (se recomienda notación hexadecimal).

Si el canal alfa (A) en el valor del color se establece en 0xFF (es decir, el byte más alto es FF), el fondo del gráfico se tratará como transparente, lo que significa que no sobrescribirá los píxeles subyacentes durante operaciones como blit.

Nota: El valor de `gráfico` no debe ser 0.

2278 dimgrob_c (hprime) — Redimensiona `gráfico` a las dimensiones especificadas `ancho` y `alto` medidas en unidades cartesianas,...

ID 2278 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: dimgrob_c (hprime)

Explicación española: Redimensiona `gráfico` a las dimensiones especificadas `ancho` y `alto` medidas en unidades cartesianas,...

Sintaxis

dimgrob_c(gráfico, ancho, alto, color)

Redimensiona gráfico a las dimensiones especificadas ancho y alto medidas en unidades cartesianas, y lo llena con el color especificado.

El color debe ser un número entero de 32 bits en formato AARRGGBB (se recomienda notación hexadecimal).

Si el canal alfa (A) en el valor del color se establece en 0xFF (es decir, el byte más alto es FF), el fondo del gráfico se tratará como transparente, lo que significa que no sobrescribirá los píxeles subyacentes durante operaciones como blit.

Nota: El valor de gráfico no debe ser 0.

2263 getpix (hprime) — Recupera el color del píxel del `gráfico` especificado en las coordenadas (`x`, `y`) y lo devuelve como...

ID 2263 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: getpix (hprime)

Explicación española: Recupera el color del píxel del `gráfico` especificado en las coordenadas (`x`, `y`) y lo devuelve como...

Sintaxis

getpix(gráfico, x, y)

Recupera el color del píxel del gráfico especificado en las coordenadas (x, y) y lo devuelve como un valor entero de 24 bits en formato RGB888.

2279 getpix_c (hprime) — Recupera el color del píxel del `gráfico` especificado en las coordenadas cartesianas (`x`, `y`) y lo...

ID 2279 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: getpix_c (hprime)

Explicación española: Recupera el color del píxel del `gráfico` especificado en las coordenadas cartesianas (`x`, `y`) y lo...

Sintaxis

getpix_c(gráfico, x, y)

Recupera el color del píxel del gráfico especificado en las coordenadas cartesianas (x, y) y lo devuelve como un valor entero de 24 bits en formato RGB888.

2264 grobw (hprime) — Devuelve el ancho en píxeles del gráfico especificado.

ID 2264 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: grobw (hprime)

Explicación española: Devuelve el ancho en píxeles del gráfico especificado.

Sintaxis

grobw(gráfico)

Devuelve el ancho en píxeles del gráfico especificado.

2280 grobw_c (hprime) — Devuelve el ancho en unidades cartesianas del gráfico especificado.

ID 2280 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: grobw_c (hprime)

Explicación española: Devuelve el ancho en unidades cartesianas del gráfico especificado.

Sintaxis

grobw_c(gráfico)

Devuelve el ancho en unidades cartesianas del gráfico especificado.

2265 grobh (hprime) — Devuelve la altura en píxeles del gráfico especificado.

ID 2265 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: grobh (hprime)

Explicación española: Devuelve la altura en píxeles del gráfico especificado.

Sintaxis

grobh(gráfico)

Devuelve la altura en píxeles del gráfico especificado.

2281 grobh_c (hprime) — Devuelve la altura en unidades cartesianas del gráfico especificado.

ID 2281 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: grobh_c (hprime)

Explicación española: Devuelve la altura en unidades cartesianas del gráfico especificado.

Sintaxis

grobh_c(gráfico)

Devuelve la altura en unidades cartesianas del gráfico especificado.

2266 line (hprime) — Dibuja una línea en el `gráfico` especificado desde (`x1`, `y1`) hasta (`x2`, `y2`) utilizando el...

ID 2266 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: line (hprime)

Explicación española: Dibuja una línea en el `gráfico` especificado desde (`x1`, `y1`) hasta (`x2`, `y2`) utilizando el...

Sintaxis

line(gráfico, x1, y1, x2, y2, color)

Dibuja una línea en el gráfico especificado desde (x1, y1) hasta (x2, y2) utilizando el color especificado.

2282 line_c (hprime) — Dibuja una línea en el `gráfico` especificado desde las coordenadas cartesianas (`x1`, `y1`) hasta...

ID 2282 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: line_c (hprime)

Explicación española: Dibuja una línea en el `gráfico` especificado desde las coordenadas cartesianas (`x1`, `y1`) hasta...

Sintaxis

line_c(gráfico, x1, y1, x2, y2, color)

Dibuja una línea en el gráfico especificado desde las coordenadas cartesianas (x1, y1) hasta (x2, y2) utilizando el color especificado.

2267 rect (hprime) — Dibuja el contorno de un rectángulo en el `gráfico` especificado en (`x`, `y`) con las dimensiones dadas...

ID 2267 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: rect (hprime)

Explicación española: Dibuja el contorno de un rectángulo en el `gráfico` especificado en (`x`, `y`) con las dimensiones dadas...

Sintaxis

rect(gráfico, x, y, ancho, alto, color)

Dibuja el contorno de un rectángulo en el gráfico especificado en (x, y) con las dimensiones dadas ancho y alto, utilizando el color especificado.

2283 rect_c (hprime) — Dibuja el contorno de un rectángulo en el `gráfico` especificado en las coordenadas cartesianas (`x`, ...

ID 2283 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: rect_c (hprime)

Explicación española: Dibuja el contorno de un rectángulo en el `gráfico` especificado en las coordenadas cartesianas (`x`, ...

Sintaxis

rect_c(gráfico, x, y, ancho, alto, color)

Dibuja el contorno de un rectángulo en el gráfico especificado en las coordenadas cartesianas (x, y) con las dimensiones dadas ancho y alto medidas en unidades cartesianas, utilizando el color especificado.

2268 fillrect (hprime) — Dibuja un rectángulo en el `gráfico` especificado en (`x`, `y`) con las dimensiones especificadas...

ID 2268 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: fillrect (hprime)

Explicación española: Dibuja un rectángulo en el `gráfico` especificado en (`x`, `y`) con las dimensiones especificadas...

Sintaxis

fillrect(gráfico, x, y, ancho, alto, color_contorno, color_relleno)

Dibuja un rectángulo en el gráfico especificado en (x, y) con las dimensiones especificadas ancho y alto, utilizando color_contorno para el borde y color_relleno para el relleno.

2284 fillrect_c (hprime) — Dibuja un rectángulo en el `gráfico` especificado en las coordenadas cartesianas (`x`, `y`) con las...

ID 2284 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: fillrect_c (hprime)

Explicación española: Dibuja un rectángulo en el `gráfico` especificado en las coordenadas cartesianas (`x`, `y`) con las...

Sintaxis

fillrect_c(gráfico, x, y, ancho, alto, color_contorno, color_relleno)

Dibuja un rectángulo en el gráfico especificado en las coordenadas cartesianas (x, y) con las dimensiones especificadas ancho y alto medidas en unidades cartesianas, utilizando color_contorno para el borde y color_relleno para el relleno.

2269 textout (hprime) — Muestra el `texto` especificado en el `gráfico` en las coordenadas (`x`, `y`) utilizando el `color` ...

Referencia traducidaIdentificador técnico: `textout` (hprime)

Explicación española: Muestra el `texto` especificado en el `gráfico` en las coordenadas (`x`, `y`) utilizando el `color`...

Sintaxis`textout(gráfico, x, y, texto, color)`

Muestra el texto especificado en el gráfico en las coordenadas (x, y) utilizando el color especificado.

2285 `textout_c` (hprime) — Muestra el texto especificado en el gráfico en las coordenadas cartesianas (x, y) utilizando el color...**Referencia traducida**Identificador técnico: `textout_c` (hprime)

Explicación española: Muestra el texto especificado en el gráfico en las coordenadas cartesianas (x, y) utilizando el color...

Sintaxis`textout_c(gráfico, x, y, texto, color)`

Muestra el texto especificado en el gráfico en las coordenadas cartesianas (x, y) utilizando el color especificado.

2286 `keyboard` (hprime) — Devuelve un número que representa el estado de cada tecla en el teclado.**Referencia traducida**Identificador técnico: `keyboard` (hprime)

Explicación española: Devuelve un número que representa el estado de cada tecla en el teclado.

Sintaxis`keyboard()`

Devuelve un número que representa el estado de cada tecla en el teclado.

Las teclas están numeradas de arriba a la izquierda (Apps) a abajo a la derecha (+).

Si la n-ésima tecla está presionada, entonces el n-ésimo bit en el número devuelto por `keyboard()` estará establecido. Por lo tanto, puede probar si una tecla está presionada o no probando el n-ésimo bit:

```
if hprime.keyboard() & (1<<15):
    print('Vars está presionado')
```

2287 `mouse` (hprime) — Devuelve una tupla de 2 tuplas que representan los 2 puntos de contacto del mouse.**Referencia traducida**Identificador técnico: `mouse` (hprime)

Explicación española: Devuelve una tupla de 2 tuplas que representan los 2 puntos de contacto del mouse.

Sintaxis`mouse()`

Devuelve una tupla de 2 tuplas que representan los 2 puntos de contacto del mouse.

Si un punto de contacto está presionado, la tupla correspondiente proporcionará las coordenadas x/y, de lo contrario, será una tupla vacía.

2288 `ticks` (hprime) — Devuelve el valor interno del reloj en milisegundos.**Referencia traducida**Identificador técnico: `ticks` (hprime)

Explicación española: Devuelve el valor interno del reloj en milisegundos.

Sintaxis

`ticks()`

Devuelve el valor interno del reloj en milisegundos.

2289 eval (hprime) — Úa la cadena como una línea de comando PPL.

ID 2289 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `eval (hprime)`

Explicación española: Úa la cadena como una línea de comando PPL.

Sintaxis

`eval(cadena)`

Evalúa la cadena como una línea de comando PPL.

Dado que las cadenas se pueden crear dinámicamente, esto le permite comunicarse de Python a PPL y enviar datos del mundo de Python al mundo de PPL, y/o utilizar funciones disponibles en la calculadora pero no en Python. El resultado de PPL se devolverá como un objeto de Python si es posible (números, complejos, cadenas), de lo contrario, se convertirá en una cadena de Python.

2290 get_cartesian (hprime) — Devuelve una lista en la forma de [xmin, xmax, ymin, ymax] que contiene los rangos de los ejes x e y...

ID 2290 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `get_cartesian (hprime)`

Explicación española: Devuelve una lista en la forma de [xmin, xmax, ymin, ymax] que contiene los rangos de los ejes x e y...

Sintaxis

`get_cartesian()`

Devuelve una lista en la forma de [xmin, xmax, ymin, ymax] que contiene los rangos de los ejes x e y para coordenadas cartesianas.

Tenga en cuenta que `get_cartesian` SOLO funciona cuando la aplicación actual es una aplicación de Python (o se basa en una aplicación de Python), de lo contrario, deberá usar la función `eval` del módulo `hprime` para obtener el valor en PPL, por ejemplo: `eval("Xmin")`.

2291 set_cartesian (hprime) — Establece los rangos de los ejes x e y para coordenadas cartesianas.

ID 2291 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `set_cartesian (hprime)`

Explicación española: Establece los rangos de los ejes x e y para coordenadas cartesianas.

Sintaxis

`set_cartesian(xmin, xmax, ymin, ymax)`

Establece los rangos de los ejes x e y para coordenadas cartesianas.

Tenga en cuenta que `get_cartesian` SOLO funciona cuando la aplicación actual es una aplicación de Python.

2081 Módulo linalg de Python

ID 2081 · título recuperado de la página española

El módulo `linalg` proporciona una colección de comandos para operaciones comunes de álgebra lineal y matrices.

La documentación de este módulo se refiere a vectores y matrices. Un vector se almacena como una lista y una matriz como una lista de listas, por lo que puede crear un vector o matriz directamente utilizando delimitadores de corchetes `[]` o `[[[]]]`.

Una matriz `x` tiene `shape(x) = (n, m)` y `size(x) = n*m`.

Un vector `x` tiene `shape(x) = (n, None)` y tamaño `n`.

Para aplicar una función a un vector o matriz, por ejemplo `math.sin(x)`, use el comando `apply()`.

Para crear un vector o matriz a partir de una función o secuencia de valores, use el comando `matrix()`.

2082 pi (linalg) — Devuelve el valor de pi utilizado por el módulo linalg (3.14159265358979).

ID 2082 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `pi` (`linalg`)

Explicación española: Devuelve el valor de pi utilizado por el módulo linalg (3.14159265358979).

Sintaxis

`pi`

Devuelve el valor de pi utilizado por el módulo linalg (3.14159265358979).

2083 matrix (linalg) — Devuelve una matriz de (n, m) con elementos contruidos a partir de f, que puede ser una función o una...

ID 2083 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `matrix` (`linalg`)

Explicación española: Devuelve una matriz de (n, m) con elementos contruidos a partir de f, que puede ser una función o una...

Sintaxis

`matrix(n, m, f)`
`matrix(n, f)`

Devuelve una matriz de (n, m) con elementos contruidos a partir de f, que puede ser una función o una lista de valores. Si f es una función, el valor devuelto coincide con `[[f(i, j) for j in range(m)] for i in range(n)]`.

Sin m, devuelve una lista contruida a partir de f, que debe ser una función. El valor devuelto coincide con `[f(i) for i in range(n)]`.

Ejemplos

```
def f(i, j): return 1/(i+j+1)
matrix(2,2,f)
```

```
matrix(2,2,[1,2,3,4])
```

```
def f(x): return x*x
matrix(5,f)
```

2084 arange (linalg) — Con un solo argumento entero, devuelve una secuencia de valores de [0, a-1].

ID 2084 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `arange` (`linalg`)

Explicación española: Con un solo argumento entero, devuelve una secuencia de valores de [0, a-1].

Sintaxis

`arange(a [, b [, c]])`

Con un solo argumento entero, devuelve una secuencia de valores de [0, a-1].

De lo contrario, devuelve un vector de valores espaciados uniformemente en el intervalo [a, b] determinado por el tamaño del paso c, que tiene un valor predeterminado de 1.

2085 linspace (linalg) — Devuelve un vector de n muestras espaciadas uniformemente calculadas en el intervalo [a, b].

ID 2085 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `linspace` (`linalg`)

Explicación española: Devuelve un vector de n muestras espaciadas uniformemente calculadas en el intervalo [a, b].

Sintaxis

```
linspace(a, b, n)
```

Devuelve un vector de n muestras espaciadas uniformemente calculadas en el intervalo [a, b].

2086 dot (linalg) — Devuelve el producto punto de a y b.

ID 2086 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: dot (linalg)

Explicación española: Devuelve el producto punto de a y b.

Sintaxis

```
dot(a, b)
```

Devuelve el producto punto de a y b.

2087 cross (linalg) — Devuelve el producto cruzado de x e y, que son vectores de tamaño 3.

ID 2087 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: cross (linalg)

Explicación española: Devuelve el producto cruzado de x e y, que son vectores de tamaño 3.

Sintaxis

```
cross(x, y)
```

Devuelve el producto cruzado de x e y, que son vectores de tamaño 3.

2088 peval (linalg) — Dado un vector p que contiene los coeficientes de un polinomio (en orden descendente) y un valor x,...

ID 2088 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: peval (linalg)

Explicación española: Dado un vector p que contiene los coeficientes de un polinomio (en orden descendente) y un valor x,...

Sintaxis

```
peval(p, x)
```

Dado un vector p que contiene los coeficientes de un polinomio (en orden descendente) y un valor x, devuelve el valor del polinomio evaluado en x utilizando el método de Horner.

Sinónimos: horner

2089 mul (linalg) — Devuelve el producto de matrices o el producto matriz-escalar de A y B.

ID 2089 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: mul (linalg)

Explicación española: Devuelve el producto de matrices o el producto matriz-escalar de A y B.

Sintaxis

```
mul(A, B)
```

Devuelve el producto de matrices o el producto matriz-escalar de A y B.

2090 transpose (linalg) — Devuelve la transpuesta de la matriz x (sin conjugación compleja).

ID 2090 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: transpose (linalg)

Explicación española: Devuelve la transpuesta de la matriz x (sin conjugación compleja).

Sintaxis

```
transpose(x)
```

Devuelve la transpuesta de la matriz x (sin conjugación compleja).

2091 solve (linalg) — Devuelve un vector x que satisface el sistema de ecuaciones lineales $Ax=b$.

ID 2091 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `solve (linalg)`

Explicación española: Devuelve un vector x que satisface el sistema de ecuaciones lineales $Ax=b$.

Sintaxis

```
solve(A, b)
```

Devuelve un vector x que satisface el sistema de ecuaciones lineales $Ax=b$.

2092 rref (linalg) — Devuelve la forma escalonada reducida de una matriz rectangular.

ID 2092 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `rref (linalg)`

Explicación española: Devuelve la forma escalonada reducida de una matriz rectangular.

Sintaxis

```
rref(x)
```

Devuelve la forma escalonada reducida de una matriz rectangular.

2093 eigenvects (linalg) — Devuelve los vectores propios de una matriz cuadrada.

ID 2093 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `eigenvects (linalg)`

Explicación española: Devuelve los vectores propios de una matriz cuadrada.

Sintaxis

```
eigenvects(x)
```

Devuelve los vectores propios de una matriz cuadrada.

Sinónimos: `egv`

2094 eig (linalg) — Devuelve los vectores propios y los valores propios de una matriz cuadrada x como una tupla...

ID 2094 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `eig (linalg)`

Explicación española: Devuelve los vectores propios y los valores propios de una matriz cuadrada x como una tupla...

Sintaxis

```
eig(x)
```

Devuelve los vectores propios y los valores propios de una matriz cuadrada x como una tupla (`eigenvects(x)`, `eigenvalues()`)

2095 inv (linalg) — Dada una matriz cuadrada no singular, devuelve una matriz que contiene la inversa de x , satisfaciendo...

ID 2095 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `inv (linalg)`

Explicación española: Dada una matriz cuadrada no singular, devuelve una matriz que contiene la inversa de x, satisfaciendo...

Sintaxis

`inv(x)`

Dada una matriz cuadrada no singular, devuelve una matriz que contiene la inversa de x, satisfaciendo $\text{mul}(x, \text{inv}(x)) = \text{mul}(\text{inv}(x), x) = \text{eye}(\text{shape}(x)[0])$.

2096 pcoeff (linalg) — Dado un vector que contiene las raíces de un polinomio, devuelve un vector que contiene los coeficientes...

ID 2096 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `pcoeff (linalg)`

Explicación española: Dado un vector que contiene las raíces de un polinomio, devuelve un vector que contiene los coeficientes...

Sintaxis

`pcoeff(x)`

Dado un vector que contiene las raíces de un polinomio, devuelve un vector que contiene los coeficientes (en orden decreciente) del polinomio univariado que tiene esas raíces.

2097 fft (linalg) — Devuelve la transformada discreta de Fourier de un vector.

ID 2097 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `fft (linalg)`

Explicación española: Devuelve la transformada discreta de Fourier de un vector.

Sintaxis

`fft(v)`

Devuelve la transformada discreta de Fourier de un vector.

2098 ifft (linalg) — Devuelve la transformada discreta de Fourier inversa de un vector.

ID 2098 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `ifft (linalg)`

Explicación española: Devuelve la transformada discreta de Fourier inversa de un vector.

Sintaxis

`ifft(x)`

Devuelve la transformada discreta de Fourier inversa de un vector.

2099 det (linalg) — Devuelve el determinante de una matriz cuadrada.

ID 2099 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `det (linalg)`

Explicación española: Devuelve el determinante de una matriz cuadrada.

Sintaxis

`det(x)`

Devuelve el determinante de una matriz cuadrada.

2100 proot (linalg) — Dado un vector que contiene los coeficientes de un polinomio, devuelve todas las raíces calculadas...

ID 2100 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `proot` (`linalg`)

Explicación española: Dado un vector que contiene los coeficientes de un polinomio, devuelve todas las raíces calculadas...

Sintaxis

`proot(x)`

Dado un vector que contiene los coeficientes de un polinomio, devuelve todas las raíces calculadas (puede que no funcione si las raíces no son simples).

2101 zeros (linalg) — Devuelve un vector (si $m=0$) o una matriz (si $m>0$) llena de ceros.

ID 2101 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `zeros` (`linalg`)

Explicación española: Devuelve un vector (si $m=0$) o una matriz (si $m>0$) llena de ceros.

Sintaxis

`zeros(n [, m])`

Devuelve un vector (si $m=0$) o una matriz (si $m>0$) llena de ceros.

2102 ones (linalg) — Devuelve un vector (si $m=0$) o una matriz (si $m>0$) llena de unos.

ID 2102 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `ones` (`linalg`)

Explicación española: Devuelve un vector (si $m=0$) o una matriz (si $m>0$) llena de unos.

Sintaxis

`ones(n [, m])`

Devuelve un vector (si $m=0$) o una matriz (si $m>0$) llena de unos.

2103 identity (linalg) — Devuelve una matriz identidad cuadrada.

ID 2103 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `identity` (`linalg`)

Explicación española: Devuelve una matriz identidad cuadrada.

Sintaxis

`identity(n)`

Devuelve una matriz identidad cuadrada.

Sinónimos: `eye`, `idn`

2104 rand (linalg) — Devuelve un vector de tamaño n (si $m=0$) o una matriz de forma (n, m) (si $m>0$) que contiene elementos...

ID 2104 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `rand` (`linalg`)

Explicación española: Devuelve un vector de tamaño n (si $m=0$) o una matriz de forma (n, m) (si $m>0$) que contiene elementos...

Sintaxis

`rand(n [, m])`

Devuelve un vector de tamaño n (si $m=0$) o una matriz de forma (n, m) (si $m>0$) que contiene elementos aleatorios.

Sinónimos: `ranm`, `ranv`

2105 shape (linalg) — Devuelve las dimensiones de un vector o matriz como una tupla (filas, columnas).

ID 2105 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `shape (linalg)`

Explicación española: Devuelve las dimensiones de un vector o matriz como una tupla (filas, columnas).

Sintaxis

`shape(a)`

Devuelve las dimensiones de un vector o matriz como una tupla (filas, columnas).

2106 size (linalg) — Devuelve el número de elementos en un vector o matriz como un entero.

ID 2106 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `size (linalg)`

Explicación española: Devuelve el número de elementos en un vector o matriz como un entero.

Sintaxis

`size(x)`

Devuelve el número de elementos en un vector o matriz como un entero.

2107 apply (linalg) — Aplica la función `f` elemento a elemento a `m`, donde `m` es una tupla, una lista o una matriz.

ID 2107 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `apply (linalg)`

Explicación española: Aplica la función `f` elemento a elemento a `m`, donde `m` es una tupla, una lista o una matriz.

Sintaxis

`apply(f, m [, matrix])`

Aplica la función `f` elemento a elemento a `m`, donde `m` es una tupla, una lista o una matriz.

Cuando `m` es una tupla o una lista, cada elemento se procesa directamente. Cuando `m` es una matriz, se debe pasar el marcador `matrix` como tercer argumento para indicar que `m` es una estructura bidimensional y que `f` debe aplicarse a cada elemento individual de la matriz.

Esta función es útil para aplicar funciones matemáticas a todos los elementos de una estructura de datos sin escribir bucles anidados explícitos.

Ejemplos

```
from linalg import *
from math import *
x = [0, pi/2, pi, 3*pi/2]
apply(sin, x)

x = [[0, pi/2], [pi, 3*pi/2]]
apply(sin, x, matrix)
```

2108 add (linalg) — Devuelve un vector o matriz que contiene la suma de `A` y `B`, elemento por elemento.

ID 2108 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `add (linalg)`

Explicación española: Devuelve un vector o matriz que contiene la suma de `A` y `B`, elemento por elemento.

Sintaxis

`add(A, B)`

Devuelve un vector o matriz que contiene la suma de A y B, elemento por elemento.

2109 sub (linalg) — Devuelve un vector o matriz que contiene la diferencia de A y B, elemento por elemento.

ID 2109 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `sub (linalg)`

Explicación española: Devuelve un vector o matriz que contiene la diferencia de A y B, elemento por elemento.

Sintaxis

`sub(A, B)`

Devuelve un vector o matriz que contiene la diferencia de A y B, elemento por elemento.

2110 real (linalg) — Devuelve un vector o matriz que contiene la parte real de x, elemento por elemento.

ID 2110 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `real (linalg)`

Explicación española: Devuelve un vector o matriz que contiene la parte real de x, elemento por elemento.

Sintaxis

`real(x)`

Devuelve un vector o matriz que contiene la parte real de x, elemento por elemento.

Sinónimos: `re`

2111 imag (linalg) — Devuelve un vector o matriz que contiene la parte imaginaria de x, elemento por elemento.

ID 2111 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `imag (linalg)`

Explicación española: Devuelve un vector o matriz que contiene la parte imaginaria de x, elemento por elemento.

Sintaxis

`imag(x)`

Devuelve un vector o matriz que contiene la parte imaginaria de x, elemento por elemento.

Sinónimos: `im`

2112 conj (linalg) — Devuelve una matriz que contiene el conjugado complejo de x, elemento por elemento.

ID 2112 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `conj (linalg)`

Explicación española: Devuelve una matriz que contiene el conjugado complejo de x, elemento por elemento.

Sintaxis

`conj(x)`

Devuelve una matriz que contiene el conjugado complejo de x, elemento por elemento.

2113 abs (linalg) — Devuelve una matriz que contiene el valor absoluto de x, elemento por elemento.

ID 2113 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `abs (linalg)`

Explicación española: Devuelve una matriz que contiene el valor absoluto de x, elemento por elemento.

Sintaxis

`abs(x)`

Devuelve una matriz que contiene el valor absoluto de x, elemento por elemento.

2130 Módulo math de Python

ID 2130 · título recuperado de la página española

El módulo math proporciona algunas funciones matemáticas básicas para trabajar con números de punto flotante.

2148 acos (math) — Devuelve el coseno inverso de x.

ID 2148 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `acos (math)`

Explicación española: Devuelve el coseno inverso de x.

Sintaxis

`acos(x)`

Devuelve el coseno inverso de x.

2142 acosh (math) — Devuelve el coseno hiperbólico inverso de x.

ID 2142 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `acosh (math)`

Explicación española: Devuelve el coseno hiperbólico inverso de x.

Sintaxis

`acosh(x)`

Devuelve el coseno hiperbólico inverso de x.

2149 asin (math) — Devuelve el arco seno de x.

ID 2149 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `asin (math)`

Explicación española: Devuelve el arco seno de x.

Sintaxis

`asin(x)`

Devuelve el arco seno de x.

2143 asinh (math) — Devuelve el seno hiperbólico inverso de x.

ID 2143 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `asinh (math)`

Explicación española: Devuelve el seno hiperbólico inverso de x.

Sintaxis

`asinh(x)`

Devuelve el seno hiperbólico inverso de x.

2150 atan (math) — Devuelve la tangente inversa de x.

ID 2150 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `atan (math)`

Explicación española: Devuelve la tangente inversa de x.

Sintaxis

`atan(x)`

Devuelve la tangente inversa de x .

2151 atan2 (math) — Devuelve el ángulo principal del punto (x, y) , teniendo en cuenta los signos de ambos argumentos para...

ID 2151 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `atan2 (math)`

Explicación española: Devuelve el ángulo principal del punto (x, y) , teniendo en cuenta los signos de ambos argumentos para...

Sintaxis

`atan2(y, x)`

Devuelve el ángulo principal del punto (x, y) , teniendo en cuenta los signos de ambos argumentos para determinar el cuadrante correcto.

2144 atanh (math) — Devuelve la tangente hiperbólica inversa de x .

ID 2144 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `atanh (math)`

Explicación española: Devuelve la tangente hiperbólica inversa de x .

Sintaxis

`atanh(x)`

Devuelve la tangente hiperbólica inversa de x .

2152 ceil (math) — Devuelve un entero, siendo x redondeado hacia el infinito positivo.

ID 2152 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `ceil (math)`

Explicación española: Devuelve un entero, siendo x redondeado hacia el infinito positivo.

Sintaxis

`ceil(x)`

Devuelve un entero, siendo x redondeado hacia el infinito positivo.

2153 copysign (math) — Devuelve x con el signo de y .

ID 2153 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `copysign (math)`

Explicación española: Devuelve x con el signo de y .

Sintaxis

`copysign(x, y)`

Devuelve x con el signo de y .

2145 cos (math) — Devuelve el coseno de x .

ID 2145 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `cos (math)`

Explicación española: Devuelve el coseno de x .

Sintaxis

`cos(x)`

Devuelve el coseno de x .

2139 cosh (math) — Devuelve el coseno hiperbólico de x.

ID 2139 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: cosh (math)

Explicación española: Devuelve el coseno hiperbólico de x.

Sintaxis

cosh(x)

Devuelve el coseno hiperbólico de x.

2166 degrees (math) — Devuelve los radianes x convertidos a grados.

ID 2166 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: degrees (math)

Explicación española: Devuelve los radianes x convertidos a grados.

Sintaxis

degrees(x)

Devuelve los radianes x convertidos a grados.

2131 e (math) — Devuelve el valor de la constante e utilizada por el módulo matemático (2.71828182845905).

ID 2131 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: e (math)

Explicación española: Devuelve el valor de la constante e utilizada por el módulo matemático (2.71828182845905).

Sintaxis

e

Devuelve el valor de la constante e utilizada por el módulo matemático (2.71828182845905).

2167 erf (math) — Devuelve la función de error de x.

ID 2167 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: erf (math)

Explicación española: Devuelve la función de error de x.

Sintaxis

erf(x)

Devuelve la función de error de x.

2168 erfc (math) — Devuelve la función de error complementaria de x.

ID 2168 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: erfc (math)

Explicación española: Devuelve la función de error complementaria de x.

Sintaxis

erfc(x)

Devuelve la función de error complementaria de x.

2134 exp (math) — Devuelve e elevado a la potencia de x.

ID 2134 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `exp` (math)

Explicación española: Devuelve e elevado a la potencia de x.

Sintaxis

`exp(x)`

Devuelve e elevado a la potencia de x.

2135 expm1 (math) — Devuelve $\exp(x) - 1$.

ID 2135 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `expm1` (math)

Explicación española: Devuelve $\exp(x) - 1$.

Sintaxis

`expm1(x)`

Devuelve $\exp(x) - 1$.

2154 fabs (math) — Devuelve el valor absoluto de x.

ID 2154 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `fabs` (math)

Explicación española: Devuelve el valor absoluto de x.

Sintaxis

`fabs(x)`

Devuelve el valor absoluto de x.

2155 floor (math) — Devuelve un entero, siendo x redondeado hacia el infinito negativo.

ID 2155 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `floor` (math)

Explicación española: Devuelve un entero, siendo x redondeado hacia el infinito negativo.

Sintaxis

`floor(x)`

Devuelve un entero, siendo x redondeado hacia el infinito negativo.

2157 fmod (math) — Devuelve el resto de x/y.

ID 2157 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `fmod` (math)

Explicación española: Devuelve el resto de x/y.

Sintaxis

`fmod(x, y)`

Devuelve el resto de x/y.

2158 frexp (math) — Descompone un número de punto flotante en su mantisa y exponente.

ID 2158 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `frexp` (math)

Explicación española: Descompone un número de punto flotante en su mantisa y exponente.

Sintaxis`fexp(x)`

Descompone un número de punto flotante en su mantisa y exponente. El valor devuelto es la tupla (m, e) tal que $x == m * 2^{**e}$ exactamente. Si $x == 0$, la función devuelve (0.0, 0), de lo contrario, se cumple la relación $0.5 <= \text{abs}(m) < 1$.

2169 gamma (math) — Devuelve la función gamma de x.

ID 2169 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `gamma (math)`

Explicación española: Devuelve la función gamma de x.

Sintaxis`gamma(x)`

Devuelve la función gamma de x.

2161 isfinite (math) — Devuelve True si x es finito.

ID 2161 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `isfinite (math)`

Explicación española: Devuelve True si x es finito.

Sintaxis`isfinite(x)`

Devuelve True si x es finito.

2162 isinf (math) — Devuelve True si x es infinito.

ID 2162 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `isinf (math)`

Explicación española: Devuelve True si x es infinito.

Sintaxis`isinf(x)`

Devuelve True si x es infinito.

2163 isnan (math) — Devuelve True si `x` no es un número.

ID 2163 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `isnan (math)`

Explicación española: Devuelve True si `x` no es un número.

Sintaxis`isnan(x)`

Devuelve True si x no es un número.

2159 ldexp (math) — Devuelve $x * (2^{**exp})$.

ID 2159 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `ldexp (math)`

Explicación española: Devuelve $x * (2^{**exp})$.

Sintaxis

`ldexp(x, exp)`

Devuelve $x * (2^{**exp})$.

2170 lgamma (math) — Devuelve el logaritmo natural de la función gamma de x.

ID 2170 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `lgamma (math)`

Explicación española: Devuelve el logaritmo natural de la función gamma de x.

Sintaxis

`lgamma(x)`

Devuelve el logaritmo natural de la función gamma de x.

2136 log (math) — Devuelve el logaritmo natural de x.

ID 2136 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `log (math)`

Explicación española: Devuelve el logaritmo natural de x.

Sintaxis

`log(x)`

Devuelve el logaritmo natural de x.

2138 log10 (math) — Devuelve el logaritmo en base 10 de x.

ID 2138 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `log10 (math)`

Explicación española: Devuelve el logaritmo en base 10 de x.

Sintaxis

`log10(x)`

Devuelve el logaritmo en base 10 de x.

2137 log2 (math) — Devuelve el logaritmo en base 2 de x.

ID 2137 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `log2 (math)`

Explicación española: Devuelve el logaritmo en base 2 de x.

Sintaxis

`log2(x)`

Devuelve el logaritmo en base 2 de x.

2160 modf (math) — Devuelve una tupla de dos flotantes, que son las partes fraccionaria e integral de x.

ID 2160 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `modf (math)`

Explicación española: Devuelve una tupla de dos flotantes, que son las partes fraccionaria e integral de x.

Sintaxis

`modf(x)`

Devuelve una tupla de dos flotantes, que son las partes fraccionaria e integral de x . Ambos valores devueltos tienen el mismo signo que x .

2132 pi (math) — Devuelve el valor de la constante π utilizada por el módulo matemático (3.14159265358979).

ID 2132 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>pi</code> (math) Explicación española: Devuelve el valor de la constante π utilizada por el módulo matemático (3.14159265358979).
Sintaxis <code>pi</code>

Devuelve el valor de la constante π utilizada por el módulo matemático (3.14159265358979).

2156 pow (math) — Devuelve x elevado a la potencia n .

ID 2156 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>pow</code> (math) Explicación española: Devuelve x elevado a la potencia n .
Sintaxis <code>pow(x, n)</code>

Devuelve x elevado a la potencia n .

2165 radians (math) — Devuelve los grados x convertidos a radianes.

ID 2165 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>radians</code> (math) Explicación española: Devuelve los grados x convertidos a radianes.
Sintaxis <code>radians(x)</code>

Devuelve los grados x convertidos a radianes.

2146 sin (math) — Devuelve el seno de x .

ID 2146 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>sin</code> (math) Explicación española: Devuelve el seno de x .
Sintaxis <code>sin(x)</code>

Devuelve el seno de x .

2140 sinh (math) — Devuelve el seno hiperbólico de x .

ID 2140 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>sinh</code> (math) Explicación española: Devuelve el seno hiperbólico de x .
Sintaxis <code>sinh(x)</code>

Devuelve el seno hiperbólico de x .

2133 sqrt (math) — Devuelve la raíz cuadrada de x.

ID 2133 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: sqrt (math)

Explicación española: Devuelve la raíz cuadrada de x.

Sintaxis

sqrt(x)

Devuelve la raíz cuadrada de x.

2147 tan (math) — Devuelve la tangente de x.

ID 2147 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: tan (math)

Explicación española: Devuelve la tangente de x.

Sintaxis

tan(x)

Devuelve la tangente de x.

2141 tanh (math) — Devuelve la tangente hiperbólica de x.

ID 2141 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: tanh (math)

Explicación española: Devuelve la tangente hiperbólica de x.

Sintaxis

tanh(x)

Devuelve la tangente hiperbólica de x.

2164 trunc (math) — Devuelve un entero, siendo x redondeado hacia 0.

ID 2164 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: trunc (math)

Explicación española: Devuelve un entero, siendo x redondeado hacia 0.

Sintaxis

trunc(x)

Devuelve un entero, siendo x redondeado hacia 0.

2114 Módulo matplotlib de Python

ID 2114 · título recuperado de la página española

El módulo matplotlib es una implementación simplificada de la popular biblioteca matplotlib. Proporciona un conjunto conciso de comandos adaptados para funciones básicas de trazado, lo que permite a los usuarios generar fácilmente varios tipos de gráficos. El módulo admite el trazado a partir de diferentes estructuras de datos como listas y arreglos. Actualmente, solo se admite el trazado de gráficos en 2D.

2117 axis (matplotlib) — Establece el tamaño de la ventana de gráficos.

ID 2117 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: axis (matplotlib)

Explicación española: Establece el tamaño de la ventana de gráficos.

Sintaxis

```
axis(x_min, x_max, y_min, y_max)
```

Establece el tamaño de la ventana de gráficos.

2124 barplot (matplotlib) — Crea un gráfico de barras correspondiente a la lista de datos proporcionada.

ID 2124 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `barplot` (`matplotlib`)

Explicación española: Crea un gráfico de barras correspondiente a la lista de datos proporcionada.

Sintaxis

```
barplot(X)
```

Crea un gráfico de barras correspondiente a la lista de datos proporcionada.

Sinónimos: `bar`

2120 boxplot (matplotlib) — Crea un diagrama de caja a partir de la lista de datos proporcionada.

ID 2120 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `boxplot` (`matplotlib`)

Explicación española: Crea un diagrama de caja a partir de la lista de datos proporcionada.

Sintaxis

```
boxplot(X)
```

Crea un diagrama de caja a partir de la lista de datos proporcionada.

Sinónimos: `boxwhisker`

2116 clf (matplotlib) — Elimina todos los objetos geométricos de la vista de gráfico.

ID 2116 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `clf` (`matplotlib`)

Explicación española: Elimina todos los objetos geométricos de la vista de gráfico.

Sintaxis

```
clf()
```

Elimina todos los objetos geométricos de la vista de gráfico.

2121 grid (matplotlib) — Con 0 como argumento, hace que los ejes del gráfico no sean visibles ('axes = 0').

ID 2121 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `grid` (`matplotlib`)

Explicación española: Con 0 como argumento, hace que los ejes del gráfico no sean visibles ('axes = 0').

Sintaxis

```
grid([n])
```

Con 0 como argumento, hace que los ejes del gráfico no sean visibles ('axes = 0'). Sin un argumento (o con cualquier argumento que no sea 0), hace que los ejes del gráfico sean visibles ('axes = 1').

2123 histogram (matplotlib) — Crea un histograma correspondiente a la lista de datos proporcionada.

ID 2123 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `histogram` (`matplotlib`)

Explicación española: Crea un histograma correspondiente a la lista de datos proporcionada.

Sintaxis

histogram(X)

Crea un histograma correspondiente a la lista de datos proporcionada.

Sinónimos: hist

2125 linear_regression_plot (matplotlib) — Dibuja los puntos individuales así como la regresión lineal correspondiente.

ID 2125 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: linear_regression_plot (matplotlib)

Explicación española: Dibuja los puntos individuales así como la regresión lineal correspondiente.

Sintaxis

linear_regression_plot(X, Y [, color])

Dibuja los puntos individuales así como la regresión lineal correspondiente. Acepta listas de valores x e y, y un argumento de color opcional.

2119 plot (matplotlib) — Acepta listas de coordenadas X e Y.

ID 2119 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: plot (matplotlib)

Explicación española: Acepta listas de coordenadas X e Y.

Sintaxis

plot(X, Y)

Acepta listas de coordenadas X e Y. Crea un gráfico conectando puntos consecutivos con una línea recta.

2126 scatterplot (matplotlib) — Acepta listas de coordenadas X e Y.

ID 2126 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: scatterplot (matplotlib)

Explicación española: Acepta listas de coordenadas X e Y.

Sintaxis

scatterplot(X, Y)

Acepta listas de coordenadas X e Y. Crea un diagrama de dispersión (los puntos consecutivos no están conectados).

Sinónimos: scatter

2115 show (matplotlib) — Abre un gráfico que muestra todos los objetos geométricos colocados.

ID 2115 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: show (matplotlib)

Explicación española: Abre un gráfico que muestra todos los objetos geométricos colocados.

Sintaxis

show()

Abre un gráfico que muestra todos los objetos geométricos colocados.

2118 text (matplotlib) — Coloca texto en el gráfico.

ID 2118 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `text` (`matplotlib`)

Explicación española: Coloca texto en el gráfico.

Sintaxis

```
text(x, y, cadena)
```

Coloca texto en el gráfico. Se coloca una marca en (x, y) y el texto se coloca ligeramente por encima, alineado a la izquierda.

2122 vector (matplotlib) — Crea un objeto vector que comienza en (origen_x, origen_y) y apunta en la dirección (dir_x, dir_y).

ID 2122 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `vector` (`matplotlib`)

Explicación española: Crea un objeto vector que comienza en (origen_x, origen_y) y apunta en la dirección (dir_x, dir_y).

Sintaxis

```
vector(origen_x, origen_y, dir_x, dir_y)
```

Crea un objeto vector que comienza en (origen_x, origen_y) y apunta en la dirección (dir_x, dir_y).

Sinónimos: `arrow`

1968 Módulo micropython de Python

ID 1968 · título recuperado de la página española

El módulo micropython permite el acceso y control de los internos de MicroPython.

1969 const (micropython) — Se utiliza para declarar que la `expresión` es una constante para que el compilador pueda optimizarla.

ID 1969 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `const` (`micropython`)

Explicación española: Se utiliza para declarar que la `expresión` es una constante para que el compilador pueda optimizarla.

Sintaxis

```
const(expresión)
```

Se utiliza para declarar que la expresión es una constante para que el compilador pueda optimizarla. Las constantes declaradas de esta manera aún son accesibles como variables globales desde fuera del módulo en el que se declaran. Por otro lado, si una constante comienza con un guion bajo, entonces está oculta, no está disponible como variable global y no ocupa memoria durante la ejecución.

Ejemplo

```
from micropython import const
CONST_X = const(123)
CONST_Y = const(2 * CONST_X + 1)
```

1970 opt_level (micropython) — Si se proporciona un nivel, esta función establece el nivel de optimización para la compilación...

ID 1970 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `opt_level` (`micropython`)

Explicación española: Si se proporciona un nivel, esta función establece el nivel de optimización para la compilación...

Sintaxis

```
opt_level([nivel])
```

Si se proporciona un nivel, esta función establece el nivel de optimización para la compilación posterior de scripts y devuelve None. De lo contrario, devuelve el nivel de optimización actual.

El nivel de optimización controla las siguientes características de compilación:

- Aserciones: en el nivel 0, las declaraciones de aserción están habilitadas y se compilan en el bytecode; en los niveles 1 y superiores, las aserciones no se compilan.
- Variable de depuración incorporada: en el nivel 0, esta variable se expande a True; en los niveles 1 y superiores, se expande a False.
- Números de línea de código fuente: en los niveles 0, 1 y 2, los números de línea de código fuente se almacenan junto con el bytecode para que las excepciones puedan informar el número de línea en el que ocurrieron; en los niveles 3 y superiores, los números de línea no se almacenan.

El nivel de optimización predeterminado es generalmente el nivel 0.

1971 mem_info (micropython) — Imprime información sobre la memoria actualmente utilizada.

ID 1971 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `mem_info` (micropython)

Explicación española: Imprime información sobre la memoria actualmente utilizada.

Sintaxis

```
mem_info([detallado])
```

Imprime información sobre la memoria actualmente utilizada. Si se proporciona el argumento detallado, se imprime información adicional.

1972 qstr_info (micropython) — Imprime información sobre las cadenas actualmente internadas.

ID 1972 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `qstr_info` (micropython)

Explicación española: Imprime información sobre las cadenas actualmente internadas.

Sintaxis

```
qstr_info([detallado])
```

Imprime información sobre las cadenas actualmente internadas. Si se proporciona el argumento detallado, se imprime información adicional.

1973 stack_use (micropython) — Devuelve un entero que representa el uso actual de la pila nativa en bytes.

ID 1973 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `stack_use` (micropython)

Explicación española: Devuelve un entero que representa el uso actual de la pila nativa en bytes.

Sintaxis

```
stack_use()
```

Devuelve un entero que representa el uso actual de la pila nativa en bytes. El valor absoluto de esto no es particularmente útil, más bien debería usarse para calcular diferencias en el uso de la pila en diferentes puntos.

Ejemplo

```
import micropython as mp
i = 5
def recursive_function(n):
    if n == 0:
        return
    print("Iteración {}: stack_use = {} bytes".format(i-n+1, mp.stack_use()))
    recursive_function(n - 1)
```

Llamar a la función con i iteraciones
`recursive_function(i)`

1974 pystack_use (micropython) — Devuelve el uso actual de la pila de Python (pystack) en bytes.

ID 1974 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `pystack_use` (micropython)

Explicación española: Devuelve el uso actual de la pila de Python (pystack) en bytes.

Sintaxis

`pystack_use()`

Devuelve el uso actual de la pila de Python (pystack) en bytes. La pystack es un área de memoria especial que MicroPython utiliza para almacenar datos temporales específicos de las llamadas a funciones de Python, particularmente al tratar con recursión o llamadas a funciones profundas.

Ejemplo

```
import micropython as mp
i = 5
def recursive_function(n):
    if n == 0:
        return
    print("Iteración {}: pystack_use = {} bytes".format(i-n+1, mp.pystack_use()))
    recursive_function(n - 1)
```

Llamar a la función con i iteraciones
`recursive_function(i)`

1975 heap_lock (micropython) — Bloquea el montón.

ID 1975 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `heap_lock` (micropython)

Explicación española: Bloquea el montón.

Sintaxis

`heap_lock()`

Bloquea el montón. Cuando está bloqueado, no se puede realizar ninguna asignación de memoria y se generará un `MemoryError` si se intenta realizar alguna asignación en el montón. Es útil cuando desea garantizar que no se realicen asignaciones en el montón, como durante secciones de código críticas.

Ejemplo

```
from micropython import *
heap_lock()
# Código crítico que no debe asignar memoria
heap_unlock()
```

1976 heap_unlock (micropython) — Desbloquea el montón.

ID 1976 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `heap_unlock` (micropython)

Explicación española: Desbloquea el montón.

Sintaxis

`heap_unlock()`

Desbloquea el montón. Cuando el montón está bloqueado, no se puede realizar ninguna asignación de memoria y se generará un `MemoryError` si se intenta realizar alguna asignación en el montón.

1977 kbd_intr (micropython) — Establece o borra el carácter de interrupción del teclado.

ID 1977 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `kbd_intr` (micropython)

Explicación española: Establece o borra el carácter de interrupción del teclado.

Sintaxis

```
kbd_intr(chr)
```

Establece o borra el carácter de interrupción del teclado.

2183 Módulo sys de Python

ID 2183 · título recuperado de la página española

El módulo sys proporciona acceso a algunas variables utilizadas o mantenidas por el intérprete y a funciones que interactúan fuertemente con el intérprete, como los argumentos de la línea de comandos o los flujos de entrada/salida estándar.

2184 path (sys) — Una lista mutable de directorios para buscar módulos importados.

ID 2184 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: path (sys)

Explicación española: Una lista mutable de directorios para buscar módulos importados.

Sintaxis

```
path
```

Una lista mutable de directorios para buscar módulos importados.

No se usa en Prime

2185 argv (sys) — Una lista mutable de argumentos con los que se inició el programa actual.

ID 2185 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: argv (sys)

Explicación española: Una lista mutable de argumentos con los que se inició el programa actual.

Sintaxis

```
argv
```

Una lista mutable de argumentos con los que se inició el programa actual.

2186 version (sys) — La versión del lenguaje Python a la que se ajusta esta implementación, como una cadena.

ID 2186 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: version (sys)

Explicación española: La versión del lenguaje Python a la que se ajusta esta implementación, como una cadena.

Sintaxis

```
version
```

La versión del lenguaje Python a la que se ajusta esta implementación, como una cadena.

2187 version_info (sys) — La versión del lenguaje Python a la que se ajusta esta implementación, como una tupla de enteros.

ID 2187 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: version_info (sys)

Explicación española: La versión del lenguaje Python a la que se ajusta esta implementación, como una tupla de enteros.

Sintaxis

```
version_info
```

La versión del lenguaje Python a la que se ajusta esta implementación, como una tupla de enteros.

2188 implementation (sys) — Objeto con información sobre la implementación actual de Python.

ID 2188 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `implementation (sys)`

Explicación española: Objeto con información sobre la implementación actual de Python.

Sintaxis

`implementation`

Objeto con información sobre la implementación actual de Python.

2189 platform (sys) — La plataforma en la que se ejecuta MicroPython.

ID 2189 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `platform (sys)`

Explicación española: La plataforma en la que se ejecuta MicroPython.

Sintaxis

`platform`

La plataforma en la que se ejecuta MicroPython.

Devuelve 'HP Prime'.

2190 byteorder (sys) — El orden de bytes del sistema ("little" o "big").

ID 2190 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `byteorder (sys)`

Explicación española: El orden de bytes del sistema ("little" o "big").

Sintaxis

`byteorder`

El orden de bytes del sistema ("little" o "big").

2191 maxsize (sys) — Valor máximo que un tipo de entero nativo puede contener en la plataforma actual, o valor máximo...

ID 2191 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `maxsize (sys)`

Explicación española: Valor máximo que un tipo de entero nativo puede contener en la plataforma actual, o valor máximo...

Sintaxis

`maxsize`

Valor máximo que un tipo de entero nativo puede contener en la plataforma actual, o valor máximo representable por el tipo de entero de MicroPython, si es menor que el valor máximo de la plataforma (eso es el caso para los puertos de MicroPython sin soporte de enteros largos).

2192 exit (sys) — Termina el programa con un estado de salida opcional.

ID 2192 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `exit (sys)`

Explicación española: Termina el programa con un estado de salida opcional.

Sintaxis

`exit([estado])`

Termina el programa con un estado de salida opcional. Si se proporciona el argumento opcional estado, se utilizará como el código de salida del programa. Un valor de 0 indica una terminación exitosa, mientras que cualquier valor distinto de cero se considera un código de error. Si no se proporciona estado, el código de salida predeterminado es 0. Esta función lanza una excepción `SystemExit` para salir del programa.

2193 `stdin (sys)` — Flujo de entrada estándar (terminal)

ID 2193 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>stdin (sys)</code> Explicación española: Flujo de entrada estándar (terminal)
Sintaxis <code>stdin</code>

Flujo de entrada estándar (terminal)

2194 `stdout (sys)` — Flujo de salida estándar (terminal)

ID 2194 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>stdout (sys)</code> Explicación española: Flujo de salida estándar (terminal)
Sintaxis <code>stdout</code>

Flujo de salida estándar (terminal)

2195 `stderr (sys)` — Flujo de error estándar (terminal)

ID 2195 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>stderr (sys)</code> Explicación española: Flujo de error estándar (terminal)
Sintaxis <code>stderr</code>

Flujo de error estándar (terminal)

2196 `modules (sys)` — Diccionario de módulos que se han cargado en el entorno de MicroPython.

ID 2196 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>modules (sys)</code> Explicación española: Diccionario de módulos que se han cargado en el entorno de MicroPython.
Sintaxis <code>modules</code>

Diccionario de módulos que se han cargado en el entorno de MicroPython. Este atributo es útil para la depuración y la introspección, ya que permite a los desarrolladores ver qué módulos están actualmente en memoria. Tenga en cuenta que este diccionario puede no incluir módulos integrados que están implícitamente disponibles.

2197 `exc_info (sys)` — Devuelve información sobre la excepción que se está manejando actualmente.

ID 2197 · identificador de HelpEn con explicación española de HelpEs
Referencia traducida Identificador técnico: <code>exc_info (sys)</code> Explicación española: Devuelve información sobre la excepción que se está manejando actualmente.

Sintaxis

```
exc_info()
```

Devuelve información sobre la excepción que se está manejando actualmente. Esta función es útil para recuperar información detallada de la excepción durante el manejo de errores. Llamar a `exc_info()` proporciona una tupla que contiene el tipo de excepción, el valor de la excepción y un objeto de seguimiento, que se puede usar para examinar la pila de llamadas y diagnosticar más a fondo el problema.

Ejemplo

```
import sys
try:
    1 / 0
except Exception:
    exc_type, exc_value, exc_traceback = sys.exc_info()
    print("Tipo de excepción:", exc_type)
    print("Valor de la excepción:", exc_value)
    print("Seguimiento:", exc_traceback)
```

2198 `print_exception (sys)` — Imprime la excepción con un seguimiento en un objeto tipo archivo `file` (o `sys.stdout` por defecto).

ID 2198 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `print_exception (sys)`

Explicación española: Imprime la excepción con un seguimiento en un objeto tipo archivo `file` (o `sys.stdout` por defecto).

Sintaxis

```
print_exception(exc, file=sys.stdout)
```

Imprime la excepción con un seguimiento en un objeto tipo archivo `file` (o `sys.stdout` por defecto).

2024 Módulo `ucollections` de Python

ID 2024 · título recuperado de la página española

Este módulo `ucollections` implementa tipos de datos de contenedor especializados que ofrecen alternativas a los contenedores integrados de propósito general de Python.

2025 `deque`, clase (`ucollections`)

ID 2025 · título localizado en HelpEs

Sintaxis

```
deque(iterable, longitud_maxima)
```

El objeto `deque` es un contenedor similar a una lista con adiciones rápidas en el extremo derecho y eliminaciones exclusivamente desde el extremo izquierdo.

Para crear un nuevo objeto `deque`, el parámetro `iterable` debe ser una tupla vacía. El `deque` siempre está limitado a la longitud máxima especificada por `longitud_maxima`. Una vez que el `deque` está lleno, cuando se agregan nuevos elementos, los elementos antiguos se descartan automáticamente desde el extremo izquierdo.

Métodos:

- `append`: Agregar un elemento al extremo derecho del `deque`.
- `popleft`: Eliminar y devolver un elemento del extremo izquierdo del `deque`.

2026 `append (ucollections)` — Agrega el elemento `x` al extremo derecho del objeto `deque`.

ID 2026 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `append (ucollections)`

Explicación española: Agrega el elemento `x` al extremo derecho del objeto `deque`.

Sintaxis

```
append(x)
```

Agrega el elemento x al extremo derecho del objeto deque. Una vez que el deque está lleno, cuando se agregan nuevos elementos, los elementos antiguos se descartan automáticamente del extremo izquierdo.

Ejemplo

```
from collections import deque
d = deque(), 3) # longitud máxima = 3
d.append('A') # el deque contiene: ['A']
d.append('B') # el deque contiene: ['A', 'B']
d.append('C') # el deque contiene: ['A', 'B', 'C']
d.append('D') # el deque contiene: ['B', 'C', 'D']
```

2027 popleft (collections) — Elimina y devuelve un elemento del lado izquierdo del deque.

ID 2027 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: popleft (collections)

Explicación española: Elimina y devuelve un elemento del lado izquierdo del deque.

Sintaxis

popleft()

Elimina y devuelve un elemento del lado izquierdo del deque. Si no hay elementos presentes, se genera un IndexError.

Esta operación es eficiente, con una complejidad de tiempo de $O(1)$, lo que significa que se ejecuta en tiempo constante, independientemente del tamaño del deque.

Ejemplo

```
from collections import deque
d = deque(), 2)
d.append('A') # deque contiene: ['A']
d.append('B') # deque contiene: ['A', 'B']
item = d.popleft() # deque contiene: ['B']
print(item) # imprime: 'A'
```

2028 namedtuple, clase (collections)

ID 2028 · título localizado en HelpEs

Sintaxis

namedtuple(nombre_tipo, nombres_campos)

namedtuple es una función de fábrica en el módulo collections que crea subclases de tuplas con campos nombrados. Estos campos nombrados proporcionan una forma de acceder a los elementos utilizando acceso estilo atributo, mejorando la legibilidad del código y la auto-documentación. Los objetos namedtuple son inmutables, lo que significa que sus valores no se pueden cambiar después de la creación, y son eficientes en términos de uso de memoria en comparación con las clases tradicionales. Son útiles para definir estructuras de datos simples y ligeras que tienen campos nombrados, mientras que aún conservan la inmutabilidad y las características de rendimiento de las tuplas.

Los nombres_campos son una secuencia de cadenas como ['x', 'y']. Alternativamente, nombres_campos puede ser una sola cadena con cada nombre de campo separado por espacios, por ejemplo 'x y'.

Ejemplo

```
from collections import *
Person = namedtuple('Person', 'first_name last_name age')
person1 = Person('John', 'Doe', 28)
print(person1.first_name) # Salida: John
print(person1.last_name) # Salida: Doe
print(person1.age) # Salida: 28
```

2029 OrderedDict, clase (collections)

ID 2029 · título localizado en HelpEs

Sintaxis

OrderedDict([items])

Devuelve una instancia de una subclase de dict que admite los métodos habituales de dict. Un OrderedDict es un dict que recuerda el orden en que las claves se insertaron por primera vez. El parámetro opcional items representa los elementos iniciales con los que se crea el diccionario. Si una nueva entrada sobrescribe una entrada existente, la posición de inserción original permanece sin cambios. Eliminar una entrada y volver a insertarla la moverá al final.

2030 clear (collections.OrderedDict) — Elimina todos los elementos del objeto

ID 2030 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: clear (collections.OrderedDict)

Explicación española: Elimina todos los elementos del objeto

Sintaxis

```
clear()
```

Elimina todos los elementos del objeto

2031 copy (collections.OrderedDict) — Devuelve una copia superficial del objeto.

ID 2031 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: copy (collections.OrderedDict)

Explicación española: Devuelve una copia superficial del objeto.

Sintaxis

```
copy()
```

Devuelve una copia superficial del objeto.

2032 Método de clase fromkeys de Python

ID 2032 · título recuperado de la página española

Sintaxis

```
OrderedDict.fromkeys(iterable [, value])
```

Crea un nuevo diccionario con claves del iterable y valores establecidos en value.

fromkeys() es un método de clase que devuelve un nuevo diccionario. value por defecto es None. Todos los valores se refieren a una sola instancia, por lo que generalmente no tiene sentido que value sea un objeto mutable como una lista vacía. Para obtener valores distintos, use una comprensión de diccionario en su lugar.

Ejemplo de uso:

```
keys = ['a', 'b', 'c']
```

```
dictionary = OrderedDict.fromkeys(keys, 0)
```

2033 get (collections.OrderedDict) — Devuelve el valor de la clave si la clave está en el diccionario, de lo contrario devuelve el valor...

ID 2033 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: get (collections.OrderedDict)

Explicación española: Devuelve el valor de la clave si la clave está en el diccionario, de lo contrario devuelve el valor...

Sintaxis

```
get(clave [, predeterminado])
```

Devuelve el valor de la clave si la clave está en el diccionario, de lo contrario devuelve el valor predeterminado. Si no se proporciona un valor predeterminado, se establece en None, por lo que este método nunca genera un KeyError.

2034 items (collections.OrderedDict) — Devuelve una nueva vista de los elementos del diccionario (pares (clave, valor)).

ID 2034 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `items` (`ucollections.OrderedDict`)

Explicación española: Devuelve una nueva vista de los elementos del diccionario (pares (clave, valor)).

Sintaxis

`items()`

Devuelve una nueva vista de los elementos del diccionario (pares (clave, valor)). Consulte la documentación de los objetos de vista.

2035 keys (`ucollections.OrderedDict`) — Devuelve una nueva vista de las claves del diccionario

ID 2035 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `keys` (`ucollections.OrderedDict`)

Explicación española: Devuelve una nueva vista de las claves del diccionario

Sintaxis

`keys()`

Devuelve una nueva vista de las claves del diccionario

2036 pop (`ucollections.OrderedDict`) — Si la clave está en el diccionario, se elimina y se devuelve su valor; de lo contrario, se devuelve el...

ID 2036 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `pop` (`ucollections.OrderedDict`)

Explicación española: Si la clave está en el diccionario, se elimina y se devuelve su valor; de lo contrario, se devuelve el...

Sintaxis

`pop(clave [, predeterminado])`

Si la clave está en el diccionario, se elimina y se devuelve su valor; de lo contrario, se devuelve el valor predeterminado. Si no se proporciona un valor predeterminado y la clave no está en el diccionario, se genera un `KeyError`.

2037 popitem (`ucollections.OrderedDict`) — Elimina y devuelve un par (clave, valor) del diccionario.

ID 2037 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `popitem` (`ucollections.OrderedDict`)

Explicación española: Elimina y devuelve un par (clave, valor) del diccionario.

Sintaxis

`popitem()`

Elimina y devuelve un par (clave, valor) del diccionario. Los pares se devuelven en orden LIFO.

`popitem()` es útil para iterar de manera destructiva sobre un diccionario, como se usa a menudo en algoritmos de conjuntos. Si el diccionario está vacío, llamar a `popitem()` genera un `KeyError`.

2038 setdefault (`ucollections.OrderedDict`) — Si la clave está en el diccionario, devuelve su valor.

ID 2038 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `setdefault` (`ucollections.OrderedDict`)

Explicación española: Si la clave está en el diccionario, devuelve su valor.

Sintaxis

`setdefault(clave [, predeterminado])`

Si la clave está en el diccionario, devuelve su valor. Si no, inserta la clave con un valor de predeterminado y devuelve el predeterminado. El predeterminado es None por defecto.

2039 update (ucollections.OrderedDict) — Actualiza el diccionario con los pares clave/valor de otro, sobrescribiendo las claves existentes.

ID 2039 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: update (ucollections.OrderedDict)

Explicación española: Actualiza el diccionario con los pares clave/valor de otro, sobrescribiendo las claves existentes.

Sintaxis

```
update([otro])
```

Actualiza el diccionario con los pares clave/valor de otro, sobrescribiendo las claves existentes. Devuelve None.

update() acepta ya sea otro objeto diccionario o un iterable de pares clave/valor (como tuplas u otros iterables de longitud dos). Si se especifican argumentos de palabra clave, el diccionario se actualiza con esos pares clave/valor: d.update(red=1, blue=2).

2040 values (ucollections.OrderedDict) — Devuelve una nueva vista de los valores del diccionario.

ID 2040 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: values (ucollections.OrderedDict)

Explicación española: Devuelve una nueva vista de los valores del diccionario.

Sintaxis

```
values()
```

Devuelve una nueva vista de los valores del diccionario. Consulte la documentación de los objetos de vista.

Una comparación de igualdad entre una vista dict.values() y otra siempre devolverá False. Esto también se aplica al comparar dict.values() consigo mismo:

2207 Módulo uerrno de Python

ID 2207 · título recuperado de la página española

El módulo uerrno proporciona acceso a códigos de error simbólicos para la excepción OSError. Un inventario particular de códigos depende del puerto de MicroPython.

2208 errorcode (uerrno)

ID 2208 · identificador de HelpEn con explicación española de HelpEs

errorcode

Diccionario que asigna códigos de error numéricos a cadenas con códigos de error simbólicos.

2209 EPERM (uerrno) — Operación no permitida.

ID 2209 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: EPERM (uerrno)

Explicación española: Operación no permitida.

EPERM

Operación no permitida. Este error ocurre cuando se intenta realizar una operación sin suficientes privilegios o permisos. Indica que la operación no está permitida debido a restricciones de seguridad o permisos.

2210 ENOENT (uerrno) — No existe tal archivo o directorio.

ID 2210 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ENOENT (uerrno)

Explicación española: No existe tal archivo o directorio.

ENOENT

No existe tal archivo o directorio. Este error ocurre cuando no se puede encontrar un archivo o directorio solicitado. Se genera típicamente al intentar abrir o acceder a un archivo que no existe en la ruta especificada.

2211 EIO (uerrno) — Error de E/S. Este error ocurre cuando una operación de entrada/salida falla debido a un problema de...

ID 2211 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: EIO (uerrno)

Explicación española: Error de E/S. Este error ocurre cuando una operación de entrada/salida falla debido a un problema de...

EIO

Error de E/S. Este error ocurre cuando una operación de entrada/salida falla debido a un problema de hardware, un error en el sistema de archivos u otros problemas relacionados con la E/S. Puede ocurrir durante operaciones de lectura, escritura u otras operaciones de archivo.

2212 EBADF (uerrno) — Descriptor de archivo no válido.

ID 2212 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: EBADF (uerrno)

Explicación española: Descriptor de archivo no válido.

EBADF

Descriptor de archivo no válido. Este error ocurre cuando se intenta realizar una operación en un descriptor de archivo que no es válido. Un descriptor de archivo se vuelve inválido si nunca se ha abierto, no está abierto en el momento de la operación o se ha cerrado previamente.

2213 EAGAIN (uerrno) — Inténtalo de nuevo.

ID 2213 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: EAGAIN (uerrno)

Explicación española: Inténtalo de nuevo.

EAGAIN

Inténtalo de nuevo. Este error indica que un recurso no está disponible temporalmente y que la operación debe ser reintentada. Ocurre cuando una operación no bloqueante no puede completarse de inmediato.

2214 ENOMEM (uerrno) — Memoria insuficiente.

ID 2214 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ENOMEM (uerrno)

Explicación española: Memoria insuficiente.

ENOMEM

Memoria insuficiente. Este error ocurre cuando una solicitud de asignación de memoria falla debido a recursos de memoria insuficientes. Indica que el sistema no puede proporcionar la cantidad solicitada de memoria al proceso. Este error se asigna típicamente a la excepción MemoryError.

2215 EACCES (uerrno) — Permiso denegado.

ID 2215 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: EACCES (uerrno)
Explicación española: Permiso denegado.

EACCES

Permiso denegado. Este error ocurre cuando se intenta realizar una operación en un archivo o recurso sin los permisos necesarios. Generalmente indica que el usuario o proceso actual carece de los derechos de acceso apropiados para realizar la operación solicitada.

2216 EEXIST (uerrno) — El archivo existe.

ID 2216 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: EEXIST (uerrno)
Explicación española: El archivo existe.

EEXIST

El archivo existe. Este error ocurre cuando se intenta crear un archivo o directorio que ya existe. Se genera típicamente en operaciones donde se pretende crear un nuevo archivo o directorio sin sobrescribir ninguna entidad existente.

2217 ENODEV (uerrno) — No existe tal dispositivo.

ID 2217 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ENODEV (uerrno)
Explicación española: No existe tal dispositivo.

ENODEV

No existe tal dispositivo. Este error ocurre cuando se intenta realizar una operación en un dispositivo que no existe o no está disponible. Indica que el dispositivo especificado no se puede acceder o no admite la operación solicitada.

2218 EISDIR (uerrno) — Es un directorio.

ID 2218 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: EISDIR (uerrno)
Explicación española: Es un directorio.

EISDIR

Es un directorio. Este error ocurre cuando se realiza una operación que espera un archivo en un directorio. Indica que la ruta especificada es un directorio, pero la operación requiere un archivo regular.

2219 EINVAL (uerrno) — Argumento no válido.

ID 2219 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: EINVAL (uerrno)
Explicación española: Argumento no válido.

EINVAL

Argumento no válido. Este error ocurre cuando se pasa un argumento no válido a una llamada al sistema o a una función. Indica que uno o más de los argumentos proporcionados no son aceptables o están fuera del rango permitido.

2220 EOPNOTSUPP (uerrno)

ID 2220 · identificador de HelpEn con explicación española de HelpEs

EOPNOTSUPP

Operación no soportada. Este error ocurre cuando se intenta realizar una operación en un socket, archivo u otro recurso que no soporta la operación. Indica que la acción solicitada no está implementada o permitida para el

recurso especificado.

2221 EADDRINUSE (uerrno)

ID 2221 · identificador de HelpEn con explicación española de HelpEs

EADDRINUSE

Dirección ya en uso. No relevante para HP Prime.

2222 ECONNABORTED (uerrno)

ID 2222 · identificador de HelpEn con explicación española de HelpEs

ECONNABORTED

Conexión abortada. No relevante para HP Prime.

2223 ECONNRESET (uerrno)

ID 2223 · identificador de HelpEn con explicación española de HelpEs

ECONNRESET

Conexión restablecida por el par. No relevante para HP Prime.

2224 ENOBUFS (uerrno) — No hay espacio de búfer disponible.

ID 2224 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ENOBUFS (uerrno)

Explicación española: No hay espacio de búfer disponible.

ENOBUFS

No hay espacio de búfer disponible. Este error ocurre cuando el sistema se queda sin espacio de búfer necesario para la asignación de recursos del sistema. Indica que hay recursos insuficientes para completar la operación solicitada, a menudo debido a una alta carga del sistema o restricciones de recursos.

2225 ENOTCONN (uerrno)

ID 2225 · identificador de HelpEn con explicación española de HelpEs

ENOTCONN

El punto final de transporte no está conectado. No relevante para HP Prime.

2226 ETIMEDOUT (uerrno)

ID 2226 · identificador de HelpEn con explicación española de HelpEs

ETIMEDOUT

Se agotó el tiempo de conexión. No relevante para HP Prime.

2227 ECONNREFUSED (uerrno)

ID 2227 · identificador de HelpEn con explicación española de HelpEs

ECONNREFUSED

Conexión rechazada. No relevante para HP Prime.

2228 EHOSTUNREACH (uerrno)

ID 2228 · identificador de HelpEn con explicación española de HelpEs

EHOSTUNREACH

Sin ruta al host. No relevante para HP Prime.

2229 EALREADY (uerrno)

ID 2229 · identificador de HelpEn con explicación española de HelpEs

EALREADY

Operación ya en progreso. No relevante para HP Prime.

2230 EINPROGRESS (uerrno)

ID 2230 · identificador de HelpEn con explicación española de HelpEs

EINPROGRESS

Operación en progreso. No relevante para HP Prime.

2241 Módulo Python uhashlib

ID 2241 · título recuperado de la página española

El módulo uhashlib implementa algoritmos de hash para datos binarios.

2242 sha256, clase (uhashlib)

ID 2242 · título localizado en HelpEs

Sintaxis

```
sha256([data])
```

Crea un objeto de hash SHA256 y opcionalmente alimenta datos (data) en él.

La clase sha256 se utiliza para calcular el hash SHA-256 de los datos (data), proporcionando un medio para generar un hash criptográfico de 256 bits (32 bytes). Esta función de hash se utiliza ampliamente para la verificación de la integridad de los datos y es crucial en operaciones criptográficas, como firmas digitales y autenticación.

Ejemplo

```
import uhashlib
```

```
hash = uhashlib.sha256()
hash.update(b"Hello, world!")
result = hash.digest()
hex_result = ""
# Convertir el resultado en hexadecimal
for b in result:
    hex_result += '%02x' % b
# Mostrar el resultado
print(hex_result)
```

2243 update (uhashlib.sha256) — Alimenta más datos binarios en el hash.

ID 2243 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `update (uhashlib.sha256)`
Explicación española: Alimenta más datos binarios en el hash.

Sintaxis

```
update(data)
```

Alimenta más datos binarios en el hash.

2244 digest (uhashlib.sha256) — Devuelve el hash de todos los datos pasados a través del hash, como un objeto de bytes.

ID 2244 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `digest (uhashlib.sha256)`
Explicación española: Devuelve el hash de todos los datos pasados a través del hash, como un objeto de bytes.

Sintaxis

```
digest()
```

Devuelve el hash de todos los datos pasados a través del hash, como un objeto de bytes. Después de llamar a este método, no se pueden agregar más datos al hash.

1982 Módulo uio de Python

ID 1982 · título recuperado de la página española

El módulo uio implementa un subconjunto del módulo CPython correspondiente.

1983 open (uio) — La función open() abre un archivo en modo texto y devuelve el objeto de archivo correspondiente...

ID 1983 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: open (uio)

Explicación española: La función open() abre un archivo en modo texto y devuelve el objeto de archivo correspondiente...

Sintaxis

```
open(filename, mode='r', buffering=-1, encoding='utf-8')
```

La función open() abre un archivo en modo texto y devuelve el objeto de archivo correspondiente (TextIOWrapper). Consulte la documentación de la función open() incorporada para obtener más detalles, ya que la función open del módulo uio es un alias de la función open del módulo builtins.

1984 FileIO, clase (uio)

ID 1984 · título localizado en HelpEs

Sintaxis

```
FileIO(filename, mode='r')
```

Abre o crea un archivo llamado según el parámetro filename, utilizando el modo definido en mode.

La clase FileIO proporciona capacidades de manejo de archivos de bajo nivel. Esta clase se utiliza principalmente para manejar archivos binarios y parcialmente archivos de texto con un control más directo sobre las operaciones de archivos que las alternativas de nivel superior. En los casos en que no se puede abrir el archivo, se genera un OSError. En el caso de archivos de texto, donde la codificación es importante, se recomienda utilizar la función open del módulo builtins.

El mode es una cadena que especifica el modo en el que se abre el archivo. El valor predeterminado es 'r' (modo de lectura). Los modos disponibles incluyen:

- 'r': Abrir para lectura (predeterminado). El archivo debe existir.
- 'w': Abrir para escritura. Si el archivo existe, se truncará (borrará). Si el archivo no existe, se creará.
- 'a': El archivo se abre para escritura con el puntero situado al final del archivo. Los nuevos datos se añaden al final del archivo.

Nota: Esta implementación de MicroPython no distingue entre modos binarios y de texto. Modos como 'rt', 'wt' y 'at' se tratan de la misma manera que sus contrapartes binarias, 'rb', 'wb' y 'ab' (o simplemente 'r', 'w' y 'a'). Los archivos siempre se tratan como archivos binarios.

1985 fileno (uio.FileIO) — Devuelve el descriptor de archivo asociado con un objeto de E/S.

ID 1985 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: fileno (uio.FileIO)

Explicación española: Devuelve el descriptor de archivo asociado con un objeto de E/S.

Sintaxis

```
fileno()
```

Devuelve el descriptor de archivo asociado con un objeto de E/S. Este método está disponible en objetos similares a archivos y se utiliza típicamente en operaciones de bajo nivel que requieren interacción directa con los descriptors de archivos del sistema operativo.

Ejemplo

```
import uio
```

```
# Abrir un archivo y obtener su descriptor de archivo
f = uio.FileIO("example.txt", "w")
```

```
fd = f.fileno()
print("Descriptor de archivo:", fd)
f.close()
```

1986 read (uio.FileIO) — El método read se utiliza para leer un número especificado de bytes (`size`) de un archivo abierto (o...

ID 1986 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: read (uio.FileIO)

Explicación española: El método read se utiliza para leer un número especificado de bytes (`size`) de un archivo abierto (o...

Sintaxis

```
read([size=-1])
```

El método read se utiliza para leer un número especificado de bytes (size) de un archivo abierto (o menos si la lectura alcanza EOF). Si se omite el parámetro size o se establece en -1, el método intenta leer todos los bytes restantes desde la posición actual del puntero de archivo hasta el final del archivo. Los bytes se devuelven en el tipo bytes.

Ejemplo

```
from uio import FileIO
with FileIO("example.bin", "rb") as file:
    # Leer un número especificado de bytes desde el inicio del archivo
    initial_bytes = file.read(5)
    # Leer todos los bytes restantes desde la posición actual hasta el final del archivo
    remaining_bytes = file.read()
    # Mostrar los bytes leídos
    print("Initial bytes:", initial_bytes)
    print("Remaining bytes:", remaining_bytes)
```

1987 readinto (uio.FileIO) — Lee bytes en un objeto de tipo bytes preasignado y escribible, especificado por el búfer (`buffer`...

ID 1987 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: readinto (uio.FileIO)

Explicación española: Lee bytes en un objeto de tipo bytes preasignado y escribible, especificado por el búfer (`buffer`...

Sintaxis

```
readinto(buffer [, size=-1])
```

Lee bytes en un objeto de tipo bytes preasignado y escribible, especificado por el búfer (buffer parámetro). Por ejemplo, buffer podría ser un bytearray. El parámetro size determina el número máximo de bytes a leer en el búfer. Si se omite size o se establece en -1, el método lee tantos bytes como el búfer puede contener. Este método devuelve el número de bytes que realmente se leyeron.

Ejemplo

```
from uio import FileIO
# Creando un búfer de tamaño específico
buffer = bytearray(10) # Tamaño del búfer de 10 bytes
with FileIO("example.bin", "rb") as file:
    # Leer bytes en el búfer
    num_bytes_read = file.readinto(buffer)
    # Mostrar el número de bytes leídos
    print("Número de bytes leídos:", num_bytes_read)
    # Mostrar el contenido del búfer
    print("Contenido del búfer:", buffer)
```

1988 readline (uio.FileIO) — El método readline lee una línea del archivo, comenzando desde la posición actual del puntero del...

ID 1988 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `readline` (`uio.FileIO`)

Explicación española: El método `readline` lee una línea del archivo, comenzando desde la posición actual del puntero del...

Sintaxis

```
readline([size=-1])
```

El método `readline` lee una línea del archivo, comenzando desde la posición actual del puntero del archivo y continuando hasta que se encuentra un carácter de nueva línea (`\n`), el final del archivo (EOF) o se alcanza el tamaño de búfer especificado (`size`). Este método es particularmente útil para leer líneas secuencialmente desde un archivo, ya que el puntero del archivo se mueve automáticamente a la posición inmediatamente después del final de la línea que se ha leído. Si se especifica `size`, solo se leerán hasta ese número de bytes. Si se omite `size` o se establece en `-1`, se leerá toda la línea hasta el carácter de nueva línea o EOF. Si se alcanza el final del archivo sin que se hayan leído bytes, se devuelve un objeto de bytes vacío (`b''`).

Ejemplo

```
from uio import FileIO
with FileIO("example.txt", "rb") as file:
    # Leer líneas del archivo
    while True:
        line = file.readline()
        if not line:
            break # Detener si se alcanza el final del archivo o no hay más datos
        print("Línea leída:", line)
```

1989 `readlines` (`uio.FileIO`) — El método `readlines` lee múltiples líneas del archivo, comenzando desde la posición actual del puntero...

ID 1989 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `readlines` (`uio.FileIO`)

Explicación española: El método `readlines` lee múltiples líneas del archivo, comenzando desde la posición actual del puntero...

Sintaxis

```
readlines()
```

El método `readlines` lee múltiples líneas del archivo, comenzando desde la posición actual del puntero del archivo hasta el final del archivo. Cada línea se lee hasta que se encuentra un carácter de nueva línea (`\n`) o EOF. Este método devuelve una lista de objetos de bytes, cada uno representando una línea del archivo. Es eficiente para leer todas las líneas en una lista, lo que puede ser útil para procesar múltiples líneas simultáneamente.

Ejemplo

```
from uio import FileIO
with FileIO("example.txt", "rb") as file:
    # Leer todas las líneas del archivo
    lines = file.readlines()
    # Procesar cada línea
    for line in lines:
        print("Línea leída:", line)
```

1990 `write` (`uio.FileIO`) — El método `write` escribe los datos especificados por el parámetro ``data`` (bytes, bytearray o string) en...

ID 1990 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `write` (`uio.FileIO`)

Explicación española: El método `write` escribe los datos especificados por el parámetro ``data`` (bytes, bytearray o string) en...

Sintaxis

```
write(data)
```

El método `write` escribe los datos especificados por el parámetro `data` (bytes, bytearray o string) en el archivo, comenzando desde la posición actual del puntero del archivo. Este método se utiliza principalmente para escribir

datos binarios en un archivo. El archivo debe abrirse en un modo que soporte escritura ('w', 'a'). El método devuelve el número de bytes escritos. Debido al almacenamiento en búfer, los datos pueden no aparecer realmente en el archivo hasta que se llame al método flush() o close().

Ejemplo

```
from uio import FileIO
with FileIO("example.bin", "wb") as file:
    data_to_write = b'Hello, world!'
    written_bytes = file.write(data_to_write)
    print("Written %d bytes." % written_bytes)
```

1991 seek (uio.FileIO) — El método seek ajusta el puntero del archivo a una ubicación específica dentro del archivo, dependiendo...

ID 1991 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: seek (uio.FileIO)

Explicación española: El método seek ajusta el puntero del archivo a una ubicación específica dentro del archivo, dependiendo...

Sintaxis

```
seek(offset [, whence=0])
```

El método seek ajusta el puntero del archivo a una ubicación específica dentro del archivo, dependiendo de los parámetros offset y whence. Esta funcionalidad es crucial para tareas que requieren acceso directo a ciertas partes de un archivo, como moverse a un byte específico para leer o establecer la posición para la próxima operación de escritura.

El offset (int) es el número de bytes para mover el puntero desde la posición especificada por whence. Puede ser positivo (para avanzar) o negativo (para retroceder). El whence (int, opcional) define el punto de partida para el offset. El valor predeterminado es 0, lo que significa que el offset se calcula desde el principio del archivo. Si se establece en 1, el offset se toma desde la posición actual del puntero del archivo. Si se establece en 2, el offset se toma desde el final del archivo, permitiendo el movimiento hacia atrás desde el final.

El método devuelve la nueva posición absoluta del puntero del archivo desde el principio del archivo. Puede generarse un OSError si ocurre un error, como al intentar posicionar el puntero fuera de los límites del archivo o cuando el archivo no está abierto para la operación requerida.

Ejemplo

```
from uio import FileIO
with FileIO("example.bin", "rb") as file:
    # Moverse al tercer byte desde el inicio
    file.seek(2)
    print("Byte en la posición 3:", file.read(1))
    # Retroceder dos bytes desde la posición actual
    file.seek(-2, 1)
    print("Byte en la posición 2:", file.read(1))
    # Ir al final del archivo
    file.seek(0, 2)
    print("Intento de lectura al final del archivo:", file.read())
```

1992 tell (uio.FileIO) — Devuelve la posición actual del puntero de archivo dentro del archivo.

ID 1992 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: tell (uio.FileIO)

Explicación española: Devuelve la posición actual del puntero de archivo dentro del archivo.

Sintaxis

```
tell()
```

Devuelve la posición actual del puntero de archivo dentro del archivo. Este método se utiliza comúnmente en el manejo de archivos para determinar el desplazamiento en bytes del puntero de archivo desde el principio del archivo. Es particularmente útil en escenarios donde necesita rastrear el progreso de las operaciones de lectura o escritura, o cuando necesita informar o registrar la posición actual de lectura/escritura para fines de diagnóstico o procesamiento de datos.

1993 flush (uio.FileIO) — El método flush se utiliza para limpiar cualquier búfer interno del archivo y garantizar que todos los...

ID 1993 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: flush (uio.FileIO)

Explicación española: El método flush se utiliza para limpiar cualquier búfer interno del archivo y garantizar que todos los...

Sintaxis

flush()

El método flush se utiliza para limpiar cualquier búfer interno del archivo y garantizar que todos los datos escritos en el archivo se escriban físicamente en el almacenamiento flash. Este método no cierra el archivo; simplemente limpia los búferes, y el archivo permanece abierto para más operaciones.

1994 close (uio.FileIO) — Cierra el archivo.

ID 1994 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: close (uio.FileIO)

Explicación española: Cierra el archivo.

Sintaxis

close()

Cierra el archivo. Una vez que el archivo está cerrado, no se puede leer ni escribir más. Cualquier operación que requiera que el archivo esté abierto generará un ValueError después de que el archivo se haya cerrado. Llamar a close() más de una vez está permitido y no causará un error, pero las llamadas posteriores no tendrán efecto.

Ejemplo

```
file = FileIO("example.bin", "wb")
try:
    file.write(b'Example data')
finally:
    file.close() # Asegúrese de que el archivo esté cerrado
try:
    file.read(10) # Esto generará un ValueError
except ValueError:
    print("Error: el archivo está cerrado.")
```

1995 TextIOWrapper, clase (uio)

ID 1995 · título localizado en HelpEs

Sintaxis

TextIOWrapper(filename, mode='r', buffering=-1, encoding='utf-8')

Abre o crea un archivo nombrado según el parámetro filename, utilizando el modo definido en mode.

Esta clase contiene métodos para operar en archivos de texto, pero no debe ser instanciada directamente. Consulte la documentación de la función incorporada open() para más detalles.

1996 fileno (uio.TextIOWrapper) — Devuelve el descriptor de archivo asociado con un objeto TextIOWrapper.

ID 1996 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: fileno (uio.TextIOWrapper)

Explicación española: Devuelve el descriptor de archivo asociado con un objeto TextIOWrapper.

Sintaxis

fileno()

Devuelve el descriptor de archivo asociado con un objeto TextIOWrapper. Este método se utiliza típicamente en operaciones de bajo nivel que requieren interacción directa con los descriptores de archivo del sistema operativo.

Ejemplo

```
# Abrir un archivo y obtener su descriptor de archivo
file = open("example.txt", "w")
fd = file.fileno()
print("Descriptor de archivo:", fd)
file.close()
```

1997 read (uio.TextIOWrapper) — El método read se utiliza para leer un número especificado de caracteres de un archivo de texto abierto...

ID 1997 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: read (uio.TextIOWrapper)

Explicación española: El método read se utiliza para leer un número especificado de caracteres de un archivo de texto abierto...

Sintaxis

```
read([size=-1])
```

El método read se utiliza para leer un número especificado de caracteres de un archivo de texto abierto (o menos si la lectura alcanza EOF). Devuelve una cadena que contiene los caracteres leídos. Si se omite el parámetro size o se establece en -1, el método intenta leer todos los caracteres restantes desde la posición actual del puntero del archivo hasta el final del archivo.

Ejemplo

```
file = open("example.txt", "r")
# Leer un número especificado de caracteres
initial_chars = file.read(5)
# Leer todos los caracteres restantes
remaining_chars = file.read()
# Mostrar la cadena leída
print("Initial chars:", initial_chars)
print("Remaining chars:", remaining_chars)
```

1998 readinto (uio.TextIOWrapper) — Lee bytes en un objeto de tipo bytes preasignado y escribible, especificado por el búfer ('buffer'...

ID 1998 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: readinto (uio.TextIOWrapper)

Explicación española: Lee bytes en un objeto de tipo bytes preasignado y escribible, especificado por el búfer ('buffer'...

Sintaxis

```
readinto(buffer [, size=-1])
```

Lee bytes en un objeto de tipo bytes preasignado y escribible, especificado por el búfer (buffer parámetro). Por ejemplo, buffer podría ser un bytearray. El parámetro size determina el número máximo de bytes a leer en el búfer. Si se omite size o se establece en -1, el método lee tantos bytes como el búfer puede contener. Este método devuelve el número de bytes que realmente se leyeron.

Ejemplo

```
# Creando un búfer de tamaño específico
buffer = bytearray(10) # Tamaño del búfer de 10 bytes
file = open("example.txt", "r")
# Leer bytes en el búfer
num_bytes_read = file.readinto(buffer)
print("Número de bytes leídos:", num_bytes_read)
print("Contenido del búfer:", buffer)
```

1999 readline (uio.TextIOWrapper) — El método readline lee una línea del archivo de texto, comenzando desde la posición actual del puntero...

ID 1999 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `readline` (`uio.TextIOWrapper`)

Explicación española: El método `readline` lee una línea del archivo de texto, comenzando desde la posición actual del puntero...

Sintaxis

```
readline([size=-1])
```

El método `readline` lee una línea del archivo de texto, comenzando desde la posición actual del puntero del archivo y continuando hasta que se encuentra un carácter de nueva línea (`\n`), el final del archivo (EOF) o el tamaño de búfer especificado por el parámetro `size`. Este método es particularmente útil para leer líneas secuencialmente desde un archivo, ya que el puntero del archivo se mueve automáticamente a la posición inmediatamente después del final de la línea que se ha leído. Si se especifica `size`, solo se leerán hasta ese número de bytes. Si se omite `size` o se establece en `-1`, se leerá toda la línea hasta el carácter de nueva línea o EOF. Al trabajar con archivos que contienen caracteres Unicode (fuera de la tabla ASCII estándar), no se recomienda usar `size` debido al riesgo de crear caracteres inconsistentes en la línea leída. Si se alcanza el final del archivo sin que se hayan leído bytes, se devuelve una cadena vacía (`""`).

Ejemplo

```
file = open("example.txt", "r")
# Leer líneas del archivo
while True:
    line = file.readline()
    if not line:
        break # Detener si se alcanza el final del archivo
    print("Línea leída:", line)
```

2000 `readlines` (`uio.TextIOWrapper`) — El método `readlines` lee múltiples líneas del archivo abierto en modo texto, comenzando desde la posición...

ID 2000 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `readlines` (`uio.TextIOWrapper`)

Explicación española: El método `readlines` lee múltiples líneas del archivo abierto en modo texto, comenzando desde la posición...

Sintaxis

```
readlines()
```

El método `readlines` lee múltiples líneas del archivo abierto en modo texto, comenzando desde la posición actual del puntero del archivo hasta el final del archivo. Cada línea se lee hasta que se encuentra un carácter de nueva línea (`\n`) o EOF. Este método devuelve una lista de objetos de cadena, cada uno representando una línea del archivo. Es eficiente para leer todas las líneas en una lista, lo que puede ser útil para procesar múltiples líneas simultáneamente.

Ejemplo

```
file = open("example.txt", "r")
# Leer todas las líneas del archivo
lines = file.readlines()
# Procesar cada línea
for line in lines:
    print("Línea leída:", line)
```

2001 `write` (`uio.TextIOWrapper`) — El método `write` escribe los datos especificados por el parámetro ``data`` (objeto `string`, `bytes` o...

ID 2001 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `write` (`uio.TextIOWrapper`)

Explicación española: El método `write` escribe los datos especificados por el parámetro ``data`` (objeto `string`, `bytes` o...

Sintaxis

```
write(data)
```

El método `write` escribe los datos especificados por el parámetro `data` (objeto `string`, `bytes` o `bytearray`) en el archivo, comenzando desde la posición actual del puntero del archivo. Este método se utiliza principalmente para escribir datos de texto en un archivo. El archivo debe abrirse en un modo que soporte escritura ('w', 'a'). El método devuelve el número de bytes escritos. Debido al almacenamiento en búfer, los datos pueden no aparecer realmente en el archivo hasta que se llame al método `flush()` o `close()`.

Ejemplo

```
file = open("example.txt", "w")
string_to_write = '¡Hola, mundo!'
written_bytes = file.write(string_to_write)
file.close()
print("Bytes escritos: %d" % written_bytes)
```

2002 seek (uio.TextIOWrapper) — El método `seek` ajusta el puntero del archivo a una ubicación específica dentro del archivo, dependiendo...

ID 2002 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `seek` (`uio.TextIOWrapper`)

Explicación española: El método `seek` ajusta el puntero del archivo a una ubicación específica dentro del archivo, dependiendo...

Sintaxis

```
seek(offset [, whence=0])
```

El método `seek` ajusta el puntero del archivo a una ubicación específica dentro del archivo, dependiendo de los parámetros `offset` y `whence`. Esta funcionalidad es crucial para tareas que requieren acceso directo a ciertas partes de un archivo, como moverse a un byte específico para leer o establecer la posición para la próxima operación de escritura.

El `offset` (int) es el número de bytes para mover el puntero desde la posición especificada por `whence`. Puede ser positivo (para avanzar) o negativo (para retroceder). El `whence` (int, opcional) define el punto de partida para el `offset`. El valor predeterminado es 0, lo que significa que el `offset` se calcula desde el principio del archivo. Si se establece en 1, el `offset` se toma desde la posición actual del puntero del archivo. Si se establece en 2, el `offset` se toma desde el final del archivo, permitiendo el movimiento hacia atrás desde el final.

El método devuelve la nueva posición absoluta del puntero del archivo desde el principio del archivo. Puede generarse un `OSError` si ocurre un error, como al intentar posicionar el puntero fuera de los límites del archivo o cuando el archivo no está abierto para la operación requerida.

Ejemplo

```
file = open("example.txt", "r")
# Moverse al tercer byte desde el inicio
file.seek(2)
print("Byte en la posición 3:", file.read(1))
# Retroceder dos bytes desde la posición actual
file.seek(-2, 1)
print("Byte en la posición 2:", file.read(1))
file.close()
```

2003 tell (uio.TextIOWrapper) — Devuelve la posición actual del puntero de archivo dentro del archivo.

ID 2003 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `tell` (`uio.TextIOWrapper`)

Explicación española: Devuelve la posición actual del puntero de archivo dentro del archivo.

Sintaxis

```
tell()
```

Devuelve la posición actual del puntero de archivo dentro del archivo. Este método se utiliza comúnmente en el manejo de archivos para determinar el desplazamiento en bytes del puntero de archivo desde el principio del archivo. Es particularmente útil en escenarios donde necesita rastrear el progreso de las operaciones de lectura o escritura, o cuando necesita informar o registrar la posición actual de lectura/escritura para fines de diagnóstico o procesamiento de datos.

2004 flush (uio.TextIOWrapper) — El método flush se utiliza para limpiar cualquier búfer interno del archivo y garantizar que todos los...

ID 2004 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: flush (uio.TextIOWrapper)

Explicación española: El método flush se utiliza para limpiar cualquier búfer interno del archivo y garantizar que todos los...

Sintaxis

flush()

El método flush se utiliza para limpiar cualquier búfer interno del archivo y garantizar que todos los datos escritos en el archivo se escriban físicamente en el almacenamiento flash. Este método no cierra el archivo; simplemente limpia los búferes, y el archivo permanece abierto para más operaciones.

2005 close (uio.TextIOWrapper) — Cierra el archivo.

ID 2005 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: close (uio.TextIOWrapper)

Explicación española: Cierra el archivo.

Sintaxis

close()

Cierra el archivo. Una vez que el archivo está cerrado, no se puede leer ni escribir más. Cualquier operación que requiera que el archivo esté abierto generará un ValueError después de que el archivo se haya cerrado. Llamar a close() más de una vez está permitido y no causará un error, pero las llamadas posteriores no tendrán efecto.

Ejemplo

```
file = open("example.txt", "w")
try:
    file.write('Cadena de ejemplo')
finally:
    file.close() # Asegúrese de que el archivo esté cerrado
try:
    file.read(10) # Esto generará ValueError
except ValueError:
    print("Error: el archivo está cerrado.")
```

2006 StringIO, clase (uio)

ID 2006 · título localizado en HelpEs

Sintaxis

StringIO([data])

La clase StringIO gestiona un búfer de cadena en memoria, funcionando de manera similar a un archivo. Le permite realizar operaciones similares a archivos en cadenas, como leer y escribir, mientras mantiene un puntero que rastrea la posición actual dentro del búfer de cadena. El objeto StringIO admite todos los métodos de archivo habituales, como read(), write(), seek(), flush() y close(). Además, está disponible el método getvalue(), que recupera el contenido actual del búfer subyacente que contiene los datos. Las operaciones en StringIO son generalmente más rápidas que las de archivos basados en disco, ya que todas las acciones ocurren en memoria. El contenido inicial del objeto StringIO se puede especificar mediante el parámetro data con un objeto string, bytes o bytearray.

2007 read (uio.StringIO) — El método read de la clase StringIO se utiliza para leer datos del búfer de cadena en memoria.

ID 2007 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: read (uio.StringIO)

Explicación española: El método read de la clase StringIO se utiliza para leer datos del búfer de cadena en memoria.

Sintaxis

```
read([size=-1])
```

El método `read` de la clase `StringIO` se utiliza para leer datos del búfer de cadena en memoria. Este método lee hasta `size` caracteres desde la posición actual del puntero, devolviendo los datos como una cadena. Si se omite el parámetro `size` o se establece en `-1`, `read` devolverá todos los caracteres desde la posición actual hasta el final del búfer.

Ejemplo

```
import uio
sio = uio.StringIO("Hello, world!")
print(sio.read(5)) # Salida: 'Hello'
print(sio.read())  # Salida: ', world!'
```

2008 `readinto` (`uio.StringIO`) — Lee bytes del búfer de cadena de origen en memoria en un objeto de tipo bytes preasignado y escribible,...

ID 2008 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `readinto` (`uio.StringIO`)

Explicación española: Lee bytes del búfer de cadena de origen en memoria en un objeto de tipo bytes preasignado y escribible,...

Sintaxis

```
readinto(buffer [, size=-1])
```

Lee bytes del búfer de cadena de origen en memoria en un objeto de tipo bytes preasignado y escribible, especificado por el búfer de destino (parámetro `buffer`). Por ejemplo, `buffer` podría ser un `bytearray`. El parámetro `size` determina el número máximo de bytes a leer en el búfer. Si se omite `size` o se establece en `-1`, el método lee tantos bytes como el búfer puede contener. Este método devuelve el número de bytes leídos.

2009 `readline` (`uio.StringIO`) — El método `readline` lee una línea del contenido del objeto `StringIO`, comenzando desde la posición actual...

ID 2009 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `readline` (`uio.StringIO`)

Explicación española: El método `readline` lee una línea del contenido del objeto `StringIO`, comenzando desde la posición actual...

Sintaxis

```
readline([size=-1])
```

El método `readline` lee una línea del contenido del objeto `StringIO`, comenzando desde la posición actual del puntero y continuando hasta un carácter de nueva línea (`\n`), el final de la cadena o el tamaño de búfer especificado (`size`). Este método es particularmente útil para leer líneas secuencialmente de una cadena, ya que el puntero se mueve automáticamente a la posición inmediatamente después del final de la línea que se ha leído. Si se especifica `size`, solo se leerán hasta ese número de bytes. Si se omite `size` o se establece en `-1`, se leerá toda la línea hasta el carácter de nueva línea o el final de la cadena. Al trabajar con cadenas que contienen caracteres Unicode (fuera de la tabla ASCII estándar), no se recomienda usar `size` debido al riesgo de crear caracteres inconsistentes en la línea leída. Si se alcanza el final de la cadena sin que se hayan leído bytes, se devuelve una cadena vacía (`""`).

2010 `write` (`uio.StringIO`) — El método `write` escribe los datos especificados por el parámetro ``data`` (`string`, `bytes` u objeto...

ID 2010 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `write` (`uio.StringIO`)

Explicación española: El método `write` escribe los datos especificados por el parámetro ``data`` (`string`, `bytes` u objeto...

Sintaxis

`write(data)`

El método `write` escribe los datos especificados por el parámetro `data` (string, bytes u objeto bytearray) en el búfer de cadena en memoria del objeto `StringIO`, comenzando desde la posición actual del puntero. Devuelve el número de bytes escritos.

2011 seek (uio.StringIO) — El método `seek` ajusta el puntero a una ubicación específica dentro del búfer `StringIO`, dependiendo de...

ID 2011 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `seek` (`uio.StringIO`)

Explicación española: El método `seek` ajusta el puntero a una ubicación específica dentro del búfer `StringIO`, dependiendo de...

Sintaxis

`seek(offset [, whence=0])`

El método `seek` ajusta el puntero a una ubicación específica dentro del búfer `StringIO`, dependiendo de los parámetros `offset` y `whence`. Esta funcionalidad es crucial para tareas que requieren acceso directo a ciertas partes del búfer, como moverse a una posición específica para leer o establecer la posición para la próxima operación de escritura.

El `offset` (int) es el número de bytes para mover el puntero desde la posición especificada por `whence`. Puede ser positivo (para avanzar) o negativo (para retroceder). El `whence` (int, opcional) define el punto de partida para el `offset`. El valor predeterminado es 0, lo que significa que el `offset` se calcula desde el principio del búfer. Si se establece en 1, el `offset` se toma desde la posición actual del puntero. Si se establece en 2, el `offset` se toma desde el final del búfer, permitiendo el movimiento hacia atrás desde el final.

El método devuelve la nueva posición absoluta del puntero desde el principio del búfer.

2012 flush (uio.StringIO) — El método `flush` en la clase `StringIO` está diseñado para ser compatible con las expectativas de una...

ID 2012 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `flush` (`uio.StringIO`)

Explicación española: El método `flush` en la clase `StringIO` está diseñado para ser compatible con las expectativas de una...

Sintaxis

`flush()`

El método `flush` en la clase `StringIO` está diseñado para ser compatible con las expectativas de una interfaz similar a un archivo, pero dado que `StringIO` gestiona datos en memoria sin ninguna operación de E/S real en un disco o almacenamiento externo, llamar a `flush` no realiza ninguna acción.

2013 close (uio.StringIO) — El método `close` en la clase `StringIO` se utiliza para indicar que el objeto ya no es necesario y para...

ID 2013 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `close` (`uio.StringIO`)

Explicación española: El método `close` en la clase `StringIO` se utiliza para indicar que el objeto ya no es necesario y para...

Sintaxis

`close()`

El método `close` en la clase `StringIO` se utiliza para indicar que el objeto ya no es necesario y para liberar cualquier recurso asociado con él. Mientras que en el manejo de archivos tradicional `close` termina las operaciones de archivo y libera recursos del sistema, para `StringIO`, que opera únicamente en memoria, este método sirve principalmente para marcar el búfer como inutilizable para más operaciones. Una vez que se llama a `close` en un objeto `StringIO`, no se debe utilizar el objeto para realizar más operaciones de lectura o escritura. Intentar hacerlo generará un error, asegurando que se mantenga la integridad de los datos y que no ocurran operaciones

inesperadas en un flujo cerrado.

2014 `getvalue (uio.StringIO)` — Recupera todo el contenido del búfer de un objeto StringIO.

ID 2014 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `getvalue (uio.StringIO)`

Explicación española: Recupera todo el contenido del búfer de un objeto StringIO.

Sintaxis

`getvalue()`

Recupera todo el contenido del búfer de un objeto StringIO. Este método es particularmente útil para acceder a todos los datos acumulados en el búfer de StringIO en cualquier momento sin afectar la posición actual del puntero.

2015 `BytesIO, clase (uio)`

ID 2015 · título localizado en HelpEs

Sintaxis

`BytesIO([data])`

La clase BytesIO gestiona un búfer de bytes en memoria, funcionando de manera similar a un archivo. Esta clase le permite realizar operaciones similares a archivos en secuencias de bytes, como leer y escribir, mientras mantiene un puntero que rastrea la posición actual dentro del búfer de bytes. El objeto BytesIO admite todos los métodos de archivo habituales, incluidos `read()`, `write()`, `seek()`, `flush()` y `close()`. Además, está disponible el método `getvalue()`, que recupera el contenido actual del búfer subyacente que contiene los datos binarios. Las operaciones en BytesIO son generalmente más rápidas que las de archivos basados en disco, ya que todas las acciones ocurren en memoria. El contenido inicial del objeto BytesIO se puede especificar con un objeto bytes, bytearray o string, lo que permite cargar inmediatamente datos binarios en el búfer.

2016 `read (uio.BytesIO)` — El método read de la clase BytesIO se utiliza para leer datos del búfer de bytes en memoria.

ID 2016 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `read (uio.BytesIO)`

Explicación española: El método read de la clase BytesIO se utiliza para leer datos del búfer de bytes en memoria.

Sintaxis

`read([size=-1])`

El método read de la clase BytesIO se utiliza para leer datos del búfer de bytes en memoria. Este método lee hasta size bytes desde la posición actual del puntero, devolviendo los datos como una secuencia de bytes. Si se omite el parámetro size o se establece en -1, read devolverá todos los bytes desde la posición actual hasta el final del búfer.

Ejemplo

```
import uio
bio = uio.BytesIO(b"Hello, world!")
print(bio.read(5)) # Salida: b'Hello'
print(bio.read()) # Salida: b', world!'
```

2017 `readinto (uio.BytesIO)` — Lee bytes en un objeto de bytes preasignado y escribible 'buffer', y devuelve el número de bytes leídos.

ID 2017 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `readinto (uio.BytesIO)`

Explicación española: Lee bytes en un objeto de bytes preasignado y escribible 'buffer', y devuelve el número de bytes leídos.

Sintaxis

`readinto(buffer [, size=-1])`

Lee bytes en un objeto de bytes preasignado y escribible buffer, y devuelve el número de bytes leídos. El parámetro size puede limitar el número de bytes leídos, y si se omite o se establece en -1, intenta llenar el búfer.

2018 readline (uio.BytesIO) — Lee una línea del contenido del objeto BytesIO, comenzando desde la posición actual del puntero y...

ID 2018 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: readline (uio.BytesIO)

Explicación española: Lee una línea del contenido del objeto BytesIO, comenzando desde la posición actual del puntero y...

Sintaxis

```
readline([size=-1])
```

Lee una línea del contenido del objeto BytesIO, comenzando desde la posición actual del puntero y continuando hasta que se alcance un carácter de nueva línea (\n), el final del búfer o el tamaño de búfer especificado (size). Si se especifica size, solo se leerán hasta esa cantidad de bytes. Si se omite o se establece en -1, se leerá toda la línea hasta el carácter de nueva línea o el final del búfer.

2019 write (uio.BytesIO) — Escribe los datos dados `data` (bytes, bytearray u objeto de cadena) en el búfer de bytes en memoria del...

ID 2019 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: write (uio.BytesIO)

Explicación española: Escribe los datos dados `data` (bytes, bytearray u objeto de cadena) en el búfer de bytes en memoria del...

Sintaxis

```
write(data)
```

Escribe los datos dados data (bytes, bytearray u objeto de cadena) en el búfer de bytes en memoria del objeto BytesIO, comenzando desde la posición actual del puntero. Devuelve el número de bytes escritos.

2020 seek (uio.BytesIO) — El método seek ajusta el puntero a una ubicación específica dentro del búfer BytesIO, dependiendo de los...

ID 2020 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: seek (uio.BytesIO)

Explicación española: El método seek ajusta el puntero a una ubicación específica dentro del búfer BytesIO, dependiendo de los...

Sintaxis

```
seek(offset [, whence=0])
```

El método seek ajusta el puntero a una ubicación específica dentro del búfer BytesIO, dependiendo de los parámetros offset y whence. Esta funcionalidad es crucial para tareas que requieren acceso directo a ciertas partes del búfer, como moverse a una posición específica para leer o establecer la posición para la próxima operación de escritura.

El offset (int) es el número de bytes para mover el puntero desde la posición especificada por whence. Puede ser positivo (para avanzar) o negativo (para retroceder). El whence (int, opcional) define el punto de partida para el offset. El valor predeterminado es 0, lo que significa que el offset se calcula desde el principio del búfer. Si se establece en 1, el offset se toma desde la posición actual del puntero. Si se establece en 2, el offset se toma desde el final del búfer, permitiendo el movimiento hacia atrás desde el final. El método devuelve la nueva posición absoluta del puntero desde el principio del búfer.

2021 flush (uio.BytesIO) — El método flush en la clase BytesIO está diseñado para ser compatible con las expectativas de una...

ID 2021 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `flush` (`uio.BytesIO`)

Explicación española: El método `flush` en la clase `BytesIO` está diseñado para ser compatible con las expectativas de una...

Sintaxis

`flush()`

El método `flush` en la clase `BytesIO` está diseñado para ser compatible con las expectativas de una interfaz similar a un archivo, pero dado que `BytesIO` gestiona datos en memoria sin ninguna operación de E/S real en un disco o almacenamiento externo, llamar a `flush` no realiza ninguna acción.

2022 `close` (`uio.BytesIO`) — El método `close` de la clase `BytesIO` se utiliza para indicar que el objeto ya no es necesario y para...

ID 2022 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `close` (`uio.BytesIO`)

Explicación española: El método `close` de la clase `BytesIO` se utiliza para indicar que el objeto ya no es necesario y para...

Sintaxis

`close()`

El método `close` de la clase `BytesIO` se utiliza para indicar que el objeto ya no es necesario y para liberar cualquier recurso asociado con él. Mientras que en el manejo de archivos tradicional `close` termina las operaciones de archivo y libera recursos del sistema, para `BytesIO`, que opera únicamente en memoria, este método sirve principalmente para marcar el búfer como inutilizable para más operaciones. Una vez que se llama a `close` en un objeto `BytesIO`, no se debe utilizar el objeto para realizar más operaciones de lectura o escritura. Intentar hacerlo generará un error, asegurando que se mantenga la integridad de los datos y que no ocurran operaciones inesperadas en un flujo cerrado.

2023 `getvalue` (`uio.BytesIO`) — Recupera todo el contenido del búfer de un objeto `BytesIO`.

ID 2023 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `getvalue` (`uio.BytesIO`)

Explicación española: Recupera todo el contenido del búfer de un objeto `BytesIO`.

Sintaxis

`getvalue()`

Recupera todo el contenido del búfer de un objeto `BytesIO`. Este método es particularmente útil para acceder a todos los datos acumulados en el búfer `BytesIO` en cualquier momento sin afectar la posición actual del puntero.

2245 Módulo `urandom` de Python

ID 2245 · título recuperado de la página española

El módulo `urandom` implementa generadores de números pseudoaleatorios para varias distribuciones.

2246 `getrandbits` (`urandom`) — Devuelve un entero aleatorio que consiste en un máximo de `k` bits, donde `k` varía de 0 a 32.

ID 2246 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `getrandbits` (`urandom`)

Explicación española: Devuelve un entero aleatorio que consiste en un máximo de `k` bits, donde `k` varía de 0 a 32.

Sintaxis

`getrandbits(k)`

Devuelve un entero aleatorio que consiste en un máximo de `k` bits, donde `k` varía de 0 a 32.

2247 seed (urandom) — Inicializa el estado interno del generador de números aleatorios.

ID 2247 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: seed (urandom)

Explicación española: Inicializa el estado interno del generador de números aleatorios.

Sintaxis

```
seed([a=None])
```

Inicializa el estado interno del generador de números aleatorios.

2248 randrange (urandom) — Devuelve un número entero aleatorio de un rango.

ID 2248 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: randrange (urandom)

Explicación española: Devuelve un número entero aleatorio de un rango.

Sintaxis

```
randrange(stop)  
randrange(start, stop [, step])
```

Devuelve un número entero aleatorio de un rango.

randrange(stop) devuelve un número entero aleatorio desde 0 hasta, pero sin incluir, stop.

randrange(start, stop) devuelve un número entero aleatorio desde start hasta, pero sin incluir, stop.

randrange(start, stop, step) devuelve un número entero aleatorio desde start hasta, pero sin incluir, stop, usando el step dado.

El resultado se selecciona de los valores generados por range(start, stop, step). Esto equivale a choice(range(start, stop, step)), pero no crea realmente un objeto range.

2249 randint (urandom) — Devuelve un entero aleatorio N tal que $a \leq N \leq b$.

ID 2249 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: randint (urandom)

Explicación española: Devuelve un entero aleatorio N tal que $a \leq N \leq b$.

Sintaxis

```
randint(a, b)
```

Devuelve un entero aleatorio N tal que $a \leq N \leq b$.

2250 choice (urandom) — Devuelve un elemento aleatorio de la secuencia no vacía `seq`.

ID 2250 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: choice (urandom)

Explicación española: Devuelve un elemento aleatorio de la secuencia no vacía `seq`.

Sintaxis

```
choice(seq)
```

Devuelve un elemento aleatorio de la secuencia no vacía seq. Si seq está vacía, se genera un IndexError.

2251 random (urandom) — Devuelve el siguiente número de punto flotante aleatorio en el rango de 0.0 a 1.0, excluyendo 1.0.

ID 2251 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `random` (`urandom`)

Explicación española: Devuelve el siguiente número de punto flotante aleatorio en el rango de 0.0 a 1.0, excluyendo 1.0.

Sintaxis

`random()`

Devuelve el siguiente número de punto flotante aleatorio en el rango de 0.0 a 1.0, excluyendo 1.0.

2252 uniform (urandom) — Devuelve un número de punto flotante aleatorio N tal que $a \leq N \leq b$ para $a \leq b$ y $b \leq N \leq a$ para $b < a$.

ID 2252 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `uniform` (`urandom`)

Explicación española: Devuelve un número de punto flotante aleatorio N tal que $a \leq N \leq b$ para $a \leq b$ y $b \leq N \leq a$ para $b < a$.

Sintaxis

`uniform(a, b)`

Devuelve un número de punto flotante aleatorio N tal que $a \leq N \leq b$ para $a \leq b$ y $b \leq N \leq a$ para $b < a$.

El valor del extremo b puede o no estar incluido en el rango dependiendo del redondeo de punto flotante en la ecuación $a + (b-a) * \text{random}()$.

2231 módulo ure de Python

ID 2231 · título recuperado de la página española

El módulo `ure` no es compatible actualmente con el HP Prime. Se recomienda usar soluciones alternativas.

2232 compile (ure) — Esta función es parte del módulo `ure`, que actualmente no es compatible con el HP Prime.

ID 2232 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `compile` (`ure`)

Explicación española: Esta función es parte del módulo `ure`, que actualmente no es compatible con el HP Prime.

Sintaxis

`compile(regex_str [, flags])`

Esta función es parte del módulo `ure`, que actualmente no es compatible con el HP Prime.

2233 match (ure) — Esta función es parte del módulo `ure`, que actualmente no es compatible con el HP Prime.

ID 2233 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `match` (`ure`)

Explicación española: Esta función es parte del módulo `ure`, que actualmente no es compatible con el HP Prime.

Sintaxis

`match(regex_str, string)`

Esta función es parte del módulo `ure`, que actualmente no es compatible con el HP Prime.

2234 search (ure) — Esta función es parte del módulo `ure`, que actualmente no es compatible con el HP Prime.

ID 2234 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `search` (`ure`)

Explicación española: Esta función es parte del módulo `ure`, que actualmente no es compatible con el HP Prime.

Sintaxis

```
search(regex_str, string)
```

Esta función es parte del módulo ure, que actualmente no es compatible con el HP Prime.

2235 DEBUG (ure) — Esta función es parte del módulo ure, que actualmente no es compatible con el HP Prime.

ID 2235 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: DEBUG (ure)

Explicación española: Esta función es parte del módulo ure, que actualmente no es compatible con el HP Prime.

Sintaxis

```
DEBUG
```

Esta función es parte del módulo ure, que actualmente no es compatible con el HP Prime.

2041 Módulo ustruct de Python

ID 2041 · título recuperado de la página española

El módulo ustruct realiza conversiones entre valores de Python y estructuras de C representadas como objetos bytes de Python. Esto se puede utilizar, entre otras cosas, para manejar datos binarios almacenados en archivos o procedentes de conexiones de red. Utiliza cadenas de formato como descripciones compactas de la disposición de las estructuras de C y de la conversión prevista a/desde valores de Python.

Caracteres de formato:

'b': signed char (1 byte)

'B': unsigned char (1 byte)

'h': signed short (2 bytes)

'H': unsigned short (2 bytes)

'i': signed int (4 bytes)

'l': signed long (4 bytes)

'I': unsigned int (4 bytes)

'L': unsigned long (4 bytes)

'q': signed long long (8 bytes)

'Q': unsigned long long (8 bytes)

'f': float (4 bytes)

'd': double (8 bytes)

2042 calcsiz (ustruct) — Devuelve el número de bytes requeridos para almacenar una estructura definida por el parámetro...

ID 2042 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: calcsiz (ustruct)

Explicación española: Devuelve el número de bytes requeridos para almacenar una estructura definida por el parámetro...

Sintaxis

```
calcsiz(format_string)
```

Devuelve el número de bytes requeridos para almacenar una estructura definida por el parámetro `format_string`. Esta cadena especifica la disposición de los tipos de datos en la estructura, indicando el tipo y el orden de los elementos de datos. Cada carácter en la cadena de formato corresponde a un tipo de datos específico con un tamaño predefinido. Es importante destacar que el cálculo del tamaño también tiene en cuenta los requisitos de alineación necesarios en arquitecturas ARM (esto también se aplica a HP Prime), asegurando que la estructura de datos se alinee con los límites de memoria apropiados para un acceso y procesamiento eficientes.

Ejemplo

```
from ustruct import calcsiz
print(calcsiz('iB')) # salida: 5
print(calcsiz('Bi')) # salida: 8
```

2043 pack (ustruct) — Empaqueta los valores de acuerdo con el parámetro ``format_string`` y devuelve un objeto de bytes que...

ID 2043 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `pack (ustruct)`

Explicación española: Empaqueta los valores de acuerdo con el parámetro ``format_string`` y devuelve un objeto de bytes que...

Sintaxis

```
pack(format_string, value_1, value_2, ...)
```

Empaqueta los valores de acuerdo con el parámetro `format_string` y devuelve un objeto de bytes que codifica estos valores.

2044 pack_into (ustruct) — Empaqueta los valores de acuerdo con el parámetro ``format_string`` en un búfer que comienza en el...

ID 2044 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `pack_into (ustruct)`

Explicación española: Empaqueta los valores de acuerdo con el parámetro ``format_string`` en un búfer que comienza en el...

Sintaxis

```
pack_into(format_string, buffer, offset, value_1, value_2, ...)
```

Empaqueta los valores de acuerdo con el parámetro `format_string` en un búfer que comienza en el desplazamiento. El valor de desplazamiento puede ser negativo para contar desde el final del búfer.

2045 unpack (ustruct) — Desempaqueta los datos según la cadena_de_formato.

ID 2045 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `unpack (ustruct)`

Explicación española: Desempaqueta los datos según la cadena_de_formato.

Sintaxis

```
unpack(cadena_de_formato, datos)
```

Desempaqueta los datos según la cadena_de_formato. El valor de retorno es una tupla de los valores desempaquetados.

2046 unpack_from (ustruct) — Desempaqueta desde ``data``, comenzando en ``offset``, según la cadena de formato especificada por...

ID 2046 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `unpack_from (ustruct)`

Explicación española: Desempaqueta desde ``data``, comenzando en ``offset``, según la cadena de formato especificada por...

Sintaxis

```
unpack_from(format_string, data [, offset=0])
```

Desempaqueta desde `data`, comenzando en `offset`, según la cadena de formato especificada por `format_string`. El parámetro `offset` indica la posición inicial y puede ser negativo para contar desde el final del búfer. El valor de retorno es una tupla con los valores desempaquetados.

2236 Módulo Python utimeq

ID 2236 · título recuperado de la página española

El módulo `utimeq` proporciona una cola de prioridad ligera basada en montículos para programar eventos basados en el tiempo. Almacena cada entrada como una tupla cuyo primer valor es el tiempo programado, seguido de los datos del evento asociado. Las entradas se ordenan por tiempo, por lo que extraer de la cola devuelve la entrada con el tiempo programado más temprano. Este módulo es útil para implementar temporizadores, planificadores y bucles de eventos.

2237 clase `utimeq` (`utimeq`)

ID 2237 · título localizado en HelpEs

Sintaxis

`utimeq(n)`

Crea una cola de montón `utimeq` con espacio para `n` entradas. Este tamaño es estático, y un intento de agregar demasiadas entradas lanzará un `IndexError` con el mensaje `'queue overflow'`.

La cola se ordena por los parámetros `time` de las entradas, y luego por el orden en que se agregaron las entradas con el mismo `time`. Por lo tanto, las entradas con parámetros `time` más bajos se extraen primero, y para las entradas con el mismo `time`, la primera que se agregó se extrae primero.

2238 `push` (`utimeq`) — Agrega una entrada a la cola de prioridad basada en montículo `utimeq`.

ID 2238 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `push` (`utimeq`)

Explicación española: Agrega una entrada a la cola de prioridad basada en montículo `utimeq`.

Sintaxis

`push(time, obj, userdata)`

Agrega una entrada a la cola de prioridad basada en montículo `utimeq`.

El parámetro `time` debe ser una marca de tiempo (consulte la función `ticks` del módulo `hprime` para obtener más detalles). Los parámetros `obj` y `userdata` no se utilizan internamente y se pueden establecer en cualquier valor.

2239 `pop` (`utimeq`) — Toma la entrada con el tiempo más bajo de la cola de heap.

ID 2239 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `pop` (`utimeq`)

Explicación española: Toma la entrada con el tiempo más bajo de la cola de heap.

Sintaxis

`pop(lista)`

Toma la entrada con el tiempo más bajo de la cola de heap. Llena los primeros tres elementos de la lista dada (que ya debe existir) con:

- El tiempo de la entrada.
- El objeto de la entrada.
- Los datos del usuario de la entrada.

Devuelve `None`.

2240 `peektime` (`utimeq`) — Devuelve el tiempo del elemento superior actual (es decir, el que se eliminará a continuación) pero sin...

ID 2240 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `peektime` (`utimeq`)

Explicación española: Devuelve el tiempo del elemento superior actual (es decir, el que se eliminará a continuación) pero sin...

Sintaxis

`peektime()`

Devuelve el tiempo del elemento superior actual (es decir, el que se eliminará a continuación) pero sin eliminarlo.

2292 Palabras clave y construcciones

ID 2292 · título localizado en HelpEs

Esta sección describe la funcionalidad de las palabras clave y las construcciones de control de flujo utilizadas en Python, que son esenciales para definir la estructura y la lógica de un programa. Estas palabras clave son reservadas, lo que significa que no se pueden usar como nombres para variables, funciones u otros identificadores.

2308 Palabra clave 'assert' de Python

ID 2308 · título recuperado de la página española

Sintaxis

`assert expresión, mensaje`

La palabra clave `assert` se utiliza para verificar que una expresión lógica dada sea verdadera. Si la expresión evalúa a `False`, se genera un `AssertionError` con un mensaje opcional. Esto es particularmente útil para fines de depuración, donde puede verificar condiciones que se espera que sean verdaderas.

La instrucción `assert` se utiliza a menudo en entornos de prueba para garantizar que el programa se comporte como se espera. Es una buena práctica utilizar instrucciones `assert` para detectar errores críticos temprano en el proceso de desarrollo.

Ejemplo

```
def apply_discount(price, discount):
    final_price = price * (1 - discount)
    assert final_price >= 0, "El precio final no debe ser negativo"
    return final_price
```

```
# Esto imprimirá el precio final si no es negativo
print(apply_discount(100, 0.2)) # Salida: 80
# Esto generará un AssertionError si se descomenta
# print(apply_discount(100, 1.5))
```

2304 break (con bucle)

ID 2304 · título localizado en HelpEs

Sintaxis

`break`

Una instrucción `break` termina el bucle que la contiene. El programa continúa con las instrucciones después del bucle. Cuando una instrucción `break` está dentro de un bucle anidado, `break` termina el bucle más interno.

Ejemplo

```
def break_demo():
    for val in "string":
        if val == "i":
            break
        print(val)
    print("break")
```

Desde la vista numérica, ejecute:

```
> break_demo()
s
t
r
break
```

2313 class (bloque class)

ID 2313 · título localizado en HelpEs

Sintaxis

`class class_name:`

Una clase (`class`) en Python es como un constructor de objetos, o un plano para crear objetos. Todas las clases tienen una función llamada `__init__()`, que siempre se ejecuta cuando se inicia la clase. Utilice la función `__init__()` para asignar valores a las propiedades del objeto, o realizar otras operaciones necesarias al crear el objeto.

Ejemplo

```
class Person:
    def __init__(self, name, age):
        self.name = name
        self.age = age
```

```
def class_demo():
    p1 = Person("John", 36)
    print(p1.name, " tiene ", p1.age, " años")
```

Desde la vista numérica, ejecute:

```
>class_demo()
John tiene 36 años
```

2305 continue (con bucle)

ID 2305 · título localizado en HelpEs

Sintaxis

continue

La instrucción continue se utiliza para omitir el resto del código dentro de un bucle solo para la iteración actual. El bucle no termina, sino que continúa con la siguiente iteración.

Ejemplo

```
def cont_demo():
    for val in "string":
        if val == "i":
            continue
        print(val)
```

Desde la vista numérica, ejecute:

```
>cont_demo()
s
t
r
n
g
```

2293 def (bloque de función)

ID 2293 · título localizado en HelpEs

Sintaxis

```
def fn_name(par1, par2, ...):
    # código
    return val
```

Definición de la función con el nombre fn_name y parámetros opcionales par1, par2, etc. para pasar a la función. La declaración de retorno es opcional para devolver un valor val de la función.

Ejemplo

```
def hello(name):
    print("Hello " + name)
```

Desde la vista numérica, ejecute:

```
>hello('World')
Hello World
```

2299 elif (bloque if-elif-else)

ID 2299 · título localizado en HelpEs

Sintaxis

```
if test1:
    declaraciones # si test1 es verdadero
elif test2:
    declaraciones # si test2 es verdadero
else:
    declaraciones # si test1 y test2 son falsos
```

La declaración `if` prueba una condición, y cuando es verdadera, ejecutará las declaraciones indentadas que siguen a la prueba. La declaración `elif` significa 'else if' y se utiliza para proporcionar condiciones de prueba adicionales. La declaración `else` sigue a la última declaración `elif` y sus declaraciones indentadas se ejecutan cuando todas las pruebas son falsas.

Ejemplo

```
def password():
    password = "123"
    attempt = input("Ingrese la contraseña: ")
    if attempt == password:
        print("Bienvenido")
    elif attempt == "":
        print("No se ingresó contraseña")
    else:
        print("Contraseña incorrecta")
```

Desde la vista numérica, ejecute:

```
>password()
```

```
Ingrese la contraseña: 1234
```

```
Contraseña incorrecta
```

2298 else (bloque if-else)

ID 2298 · título localizado en HelpEs

Sintaxis

```
if test:
    declaraciones # si verdadero
else:
    declaraciones # si falso
```

La declaración `if` prueba alguna condición, y cuando es verdadera ejecutará las declaraciones indentadas que siguen a la prueba. La declaración `else` sigue a la declaración `if` y sus declaraciones indentadas se ejecutan cuando la prueba es falsa.

Ejemplo

```
def password():
    password = "123"
    attempt = input("Ingrese la contraseña: ")
    if attempt == password:
        print("Bienvenido")
    else:
        print("Contraseña incorrecta")
```

Desde la vista numérica, ejecute:

```
>password()
```

```
Ingrese la contraseña: 1234
```

```
Contraseña incorrecta
```

2300 for (bucle de secuencia)

ID 2300 · título localizado en HelpEs

Sintaxis

```
for x in iterable:
    ...
```

Un bucle `for` itera sobre un objeto iterable, como una lista, una tupla, un diccionario, un conjunto o una cadena. Para cada elemento, el elemento se asigna a la variable objetivo y se ejecuta el bloque de código con sangría. Si el iterable está vacío, el bloque no se ejecuta.

Ejemplo

```
def for_demo():
    seq = ["One", "Two", "Three"]
    for x in seq:
        print(x)
```

Desde la vista numérica, ejecute:

```
>for_demo()
```

```
One
```

```
Two
```


Three

2301 for (bucle de rango)

ID 2301 · título localizado en HelpEs

Sintaxis

```
for i in range(min, max, step):
```

Un bucle for que utiliza `range()` itera sobre una variable entera, comenzando desde `min` y deteniéndose antes de alcanzar (exclusivamente) `max`. El bucle itera en orden ascendente o descendente dependiendo del valor de `step`. El argumento `step` determina el incremento (o decremento) con el que se actualiza la variable en cada iteración. Si `step` es positivo, el bucle avanza en orden ascendente, y si `step` es negativo, avanza en orden descendente. Cuando solo se proporcionan dos valores para `range()`, el bucle asume un `step` de 1. Si solo se proporciona un valor, `min` se establece en cero por defecto.

Ejemplo

```
def for_demo():
    for i in range(3):
        print(i)
```

Desde la vista numérica, ejecute:

```
>for_demo()
```

```
0
```

```
1
```

```
2
```

2302 else (con bucle for)

ID 2302 · título localizado en HelpEs

Sintaxis

```
for x in seq/range():
    declaraciones
    break
else:
    declaraciones
```

En un bucle for con una cláusula `else`, la cláusula `else` se ejecuta después de que el bucle se completa normalmente. La declaración `break` es útil aquí para evitar la cláusula `else`.

Ejemplo

```
def for_demo():
    for n in range(2,10):
        for x in range(2,n):
            if n % x == 0:
                print(n, " igual a ", x, "*", n/x)
                break # evitar la cláusula else
        else:
            print(n, " es un número primo")
```

Desde la vista numérica, ejecute:

```
>for_demo()
```

```
2 es un número primo
```

```
3 es un número primo
```

```
4 igual a 2 * 2.0
```

```
5 es un número primo
```

```
6 igual a 2 * 3.0
```

```
7 es un número primo
```

```
8 igual a 2 * 4.0
```

```
9 igual a 3 * 3.0
```

2310 Palabra clave 'global' de Python

ID 2310 · título recuperado de la página española

Sintaxis

```
global var
```

La palabra clave `global` permite la modificación de la variable `global var` fuera del ámbito actual. Se utiliza para crear una variable global y realizar cambios en la variable en un contexto local. Las reglas básicas para la palabra clave `global` en Python son:

- Cuando creamos una variable dentro de una función, es local por defecto.
- Cuando definimos una variable fuera de una función, es global por defecto. No es necesario usar la palabra clave `global`.
- Usamos la palabra clave `global` para leer y escribir una variable global dentro de una función.
- El uso de la palabra clave `global` fuera de una función no tiene efecto

Ejemplo

```
def add():
    global c
    c = 2
    print("c: ", c)
```

```
def add2():
    try:
        print("c: ",c)
    except NameError:
        print("c no inicializado")
```

Desde la vista numérica, ejecute:

```
>add()
```

```
c: 2
```

```
>add2()
```

```
c: 2
```

Desde la vista numérica, ejecute:

```
>add2()
```

```
c no inicializado
```

2297 if (bloque if)

ID 2297 · título localizado en HelpEs

Sintaxis

```
if test:
    statements # si es verdadero
```

La instrucción `if` prueba una condición y, cuando es `True`, ejecutará las declaraciones indentadas que siguen a la prueba.

Ejemplo

```
def password():
    password = "123"
    attempt = input("Ingrese la contraseña: ")
    if attempt == password:
        print("Bienvenido")
```

Desde la vista numérica, ejecute:

```
>password()
```

```
Ingrese la contraseña: 123
```

```
Bienvenido
```

2315 Palabra clave 'import' de Python

ID 2315 · título recuperado de la página española

Sintaxis

```
import module_name [as alias_name]
from module_name import component [as alias_name]
```

La palabra clave `'import'` se utiliza para incluir módulos externos o componentes específicos de un módulo en el espacio de nombres actual. Permite la reutilización de código de bibliotecas y otros archivos de Python.

2316 Palabra clave 'in' de Python

ID 2316 · título recuperado de la página española

Sintaxis

```
val in seq
```

La palabra clave 'in' en Python se utiliza para verificar si un valor `val` está presente en una secuencia `seq` (como una lista, una tupla, una cadena o un diccionario) o un objeto iterable. Devuelve `True` si se encuentra el valor, de lo contrario, devuelve `False`. Se utiliza comúnmente para pruebas de pertenencia e iteración a través de elementos en bucles.

Ejemplo

```
5 in [4,5,6] # True
```

2314 Palabra clave 'is' de Python

ID 2314 · título recuperado de la página española

Sintaxis

```
obj1 is obj2
```

La palabra clave 'is' se utiliza para determinar si dos objetos se refieren a la misma ubicación de memoria. Devuelve `True` si ambos objetos tienen la misma identidad (es decir, son el mismo objeto en memoria), de lo contrario, devuelve `False`. No debe confundirse con el operador `==`, que verifica la igualdad de valores.

Ejemplo

```
Ellipsis is None # False
```

2306 Expresión lambda de Python

ID 2306 · título recuperado de la página española

Sintaxis

```
lambda argumentos: expresión
```

Las expresiones lambda proporcionan una forma de crear pequeñas funciones anónimas. Estas funciones se definen utilizando la palabra clave `lambda` y están limitadas a una sola expresión. Permiten una forma más concisa de definir funciones cortas en línea sin necesidad de declararlas formalmente.

Las funciones lambda se utilizan ampliamente en situaciones donde se requiere una función simple por un corto período, particularmente en funciones de orden superior como `map()`, `filter()` y `sorted()`, donde no es necesario definir una función formal.

Ejemplo

```
numbers = [1, 2, 3, 4, 5, 6, 7]
even_numbers = list(filter(lambda x: x % 2 == 0, numbers))
print(even_numbers) # Salida: [2, 4, 6]
```

2311 Palabra clave 'nonlocal' de Python

ID 2311 · título recuperado de la página española

Sintaxis

```
nonlocal nombre_variable
```

La palabra clave `nonlocal` se utiliza en funciones anidadas para declarar que una variable no es local para la función anidada, sino que existe en el ámbito de cierre más cercano que no es global. Esto permite que la función anidada modifique la variable del ámbito de cierre. La palabra clave `nonlocal` es crucial en escenarios donde necesitas modificar una variable que fue definida en una función externa y deseas que estos cambios se reflejen en ese ámbito externo.

Esta característica es particularmente útil en funciones complejas donde es necesario mantener el estado a través de varias llamadas de funciones anidadas.

Ejemplo

```
def counter():  
    count = 0 # La variable count en la función externa
```

```
def increment():  
    nonlocal count # count de la función externa  
    count += 1  
    return count  
  
return increment
```

```
# Crear una función increment  
inc = counter()  
# Incrementar el count varias veces  
print(inc()) # Salida: 1  
print(inc()) # Salida: 2  
print(inc()) # Salida: 3
```

La palabra clave nonlocal permite que la función increment modifique la variable count definida en counter.

2309 Python 'pass' palabra clave

ID 2309 · título recuperado de la página española

Sintaxis

```
pass
```

La palabra clave pass es un marcador de posición que no hace nada cuando se ejecuta. Es especialmente útil en situaciones donde se requiere un bloque de código por la sintaxis del lenguaje (como en bucles, funciones o condicionales), pero donde no se necesita ninguna operación real.

2296 raise (instrucción)

ID 2296 · título localizado en HelpEs

Sintaxis

```
raise *Error* ("cadena")
```

La instrucción raise se utiliza para forzar una excepción especificada.

Ejemplo:

```
def divide(x,y):  
    try:  
        result = x / y  
    if result < 0:  
        raise ValueError("Resultado menor que cero")  
    except ZeroDivisionError:  
        print("División por cero")  
    except ValueError:  
        print("Hubo un ValueError")  
    else:  
        print("Resultado: ", result)
```

Desde la vista numérica, ejecute:

```
>divide(-1,2)  
Hubo un ValueError
```

2294 return (instrucción)

ID 2294 · título localizado en HelpEs

Sintaxis

```
return [resultado] # resultado es opcional
```

La instrucción return termina la ejecución de una función y devuelve un resultado a la función que la llamó. Las instrucciones que siguen a return no se ejecutan. Una instrucción return sin valor devuelve el valor None de Python.

Ejemplo

```
import math
def deg2rad(grado):
    return grado * math.pi / 180
```

```
def arc_len(radio, ángulo):
    arco = radio * deg2rad(ángulo)
    print(arco)
```

Desde la vista numérica, ejecute:

```
>arc_len(5,180)
15.70796326794897
```

2295 try (bloque try-except-else-finally)

ID 2295 · título localizado en HelpEs

Sintaxis

```
try:
    declaraciones
except *Error*:
    declaraciones # si ocurre algún error
else:
    declaraciones # si no hay error
finally:
    declaraciones # ejecutado en todas las circunstancias
```

La declaración try se utiliza para capturar excepciones que son manejadas por una o más cláusulas except. Se puede agregar una cláusula else para seguir a todas las cláusulas except y se ejecuta solo si no se produjeron excepciones. Se puede agregar una cláusula finally para ejecutarse en todas las circunstancias y puede usarse para realizar acciones de limpieza.

Ejemplo

```
def divide(x,y):
    try:
        result = x / y
    except ZeroDivisionError:
        print("División por cero")
    else:
        print("Resultado: ", result)
    finally:
        print("Fin.")
```

Desde la vista numérica, ejecute:

```
>divide(1,2)
Resultado: 0,5
Fin.
```

2303 while (bucle)

ID 2303 · título localizado en HelpEs

Sintaxis

```
while expresión:
    declaraciones # mientras la expresión sea verdadera
```

Prueba repetidamente una expresión, y si es verdadera, ejecuta las declaraciones con sangría. La primera vez que la expresión es falsa, el bucle termina. Se puede usar una cláusula else opcional para ejecutar después de la terminación cuando la expresión es falsa.

Ejemplo

```
def while_demo():
    x = 0
    while x<5:
        x += 1
        print(x)
```

Desde la vista numérica, ejecute:

```
>while_demo()
1
2
```

3
4
5

2312 Palabra clave 'with' de Python

ID 2312 · título recuperado de la página española

Sintaxis

```
with expresión [as variable]:  
    bloque_de_código
```

La palabra clave 'with' se utiliza para envolver la ejecución de un bloque de código dentro de métodos definidos por el protocolo de gestión de contexto. Asegura que los recursos se gestionen correctamente, por ejemplo, los archivos se cierran después de que se han procesado sus contenidos, o los bloqueos se liberan correctamente después de su uso. La declaración 'with' simplifica el manejo de excepciones al encapsular tareas comunes de preparación y limpieza de una manera que hace que el código sea más limpio y legible.

En contextos donde la gestión y limpieza de recursos son críticas, usar 'with' puede ayudar a prevenir fugas de recursos al asegurar que los recursos se liberen correctamente después de su uso, independientemente de si ha ocurrido un error o no.

Ejemplo

```
# Ejemplo de uso de la declaración 'with' para operaciones de archivos  
with open('example.txt', 'r') as file:  
    data = file.read()  
    print(data) # Esto imprimirá el contenido del archivo
```

El archivo se cierra automáticamente después del bloque with, incluso si ocurre un error dentro del bloque.

Para crear una clase que permita el uso de la declaración with, la clase debe implementar dos métodos especiales:

- `__enter__(self)`: Este método se llama cuando el programa entra en el bloque with. Normalmente, devuelve un objeto que se asigna a la variable después de la palabra clave as.
- `__exit__(self, exc_type, exc_value, traceback)`: Este método se llama cuando el programa sale del bloque with. Estos parámetros están relacionados con las excepciones, si es que ocurrieron. Si el método `__exit__` devuelve True, la excepción se "silencia"; de lo contrario, se propagará más adelante.

2307 Palabra clave 'yield' de Python

ID 2307 · título recuperado de la página española

Sintaxis

```
yield expresión
```

La palabra clave yield en Python se utiliza en una función para devolver un generador. Un generador es un objeto que permite la generación perezosa (a demanda) de valores durante la iteración, en lugar de devolver todos los resultados a la vez, como ocurre con return. Cuando se llama a yield, la función pausa su ejecución mientras mantiene su estado, lo que le permite reanudar desde donde se detuvo en la siguiente llamada. Esto permite una evaluación perezosa, que es más eficiente en memoria al trabajar con grandes conjuntos de datos o flujos de datos.

Ejemplo

```
def fibonacci_sequence():  
    a, b = 0, 1  
    while True:  
        yield a  
        a, b = b, a + b
```

Generador devuelto

```
gen = fibonacci_sequence()
```

Obtener los dos primeros números de Fibonacci

```
print(next(gen))
```

```
print(next(gen))
```

Iterar a través de más números de Fibonacci

```
for _ in range(5):
```

```
    print(next(gen), end = ' ')
```

```
# Salidas:  
# 0  
# 1  
# 1, 2, 3, 5, 8
```

1933 Operadores

ID 1933 · título localizado en HelpEs

Una lista detallada de los operadores de Python con ejemplos de uso.

1934 Operadores lógicos

ID 1934 · título localizado en HelpEs

Los operadores lógicos se utilizan para realizar operaciones booleanas. Evalúan expresiones para determinar relaciones lógicas, como la conjunción (and), la disyunción (or) y la negación (not). Estos operadores devuelven un valor booleano o, en algunos casos, el propio operando, dependiendo del contexto de evaluación.

2317 Operador lógico AND

ID 2317 · título recuperado de la página española

Sintaxis

a and b

Realiza una operación lógica AND. Devuelve True si ambos operandos se evalúan como True. Si el primer operando es False, el segundo operando no se evalúa (cortocircuito).

Ejemplo

```
a = True  
b = False  
result = a and b # AND lógico  
print(result) # Salidas: False
```

2318 Operador lógico OR

ID 2318 · título recuperado de la página española

Sintaxis

a or b

Realiza una operación lógica OR. Devuelve True si al menos uno de los operandos se evalúa como True. Si el primer operando es True, el segundo operando no se evalúa (cortocircuito).

Ejemplo

```
a = True  
b = False  
result = a or b # OR lógico  
print(result) # Salidas: True
```

2319 Operador lógico NOT

ID 2319 · título recuperado de la página española

Sintaxis

not a

Realiza una negación lógica. Devuelve True si el operando se evalúa como False, y False si el operando se evalúa como True.

Ejemplo

```
a = True  
result = not a # negación lógica  
print(result) # Salidas: False
```

1935 Operadores aritméticos

Los operadores aritméticos se utilizan para realizar operaciones matemáticas básicas como suma, resta, multiplicación y división. Python también admite operaciones avanzadas como la división entera, encontrar el resto y la exponenciación. Algunos de estos operadores también pueden funcionar con otros tipos de datos, como cadenas para concatenación o secuencias para repetición.

2320 Operador de adición (+)

Sintaxis

`a + b`

Realiza una operación de suma. Si los operandos son números, devuelve su suma. Si los operandos son cadenas, las concatena. Este operador también se puede usar con secuencias, como listas o tuplas, para concatenarlas.

Ejemplo

```
# Con números
a = 5
b = 3
c = a + b # suma
print(c) # Salidas: 8
```

```
# Con cadenas
x = "Hello"
y = "World"
z = x + " " + y # concatenación
print(z) # Salidas: 'Hello World'

# Con listas
list1 = [1, 2, 3]
list2 = [4, 5, 6]
result = list1 + list2 # concatenación
print(result) # Salidas: [1, 2, 3, 4, 5, 6]
```

2321 Operador de resta (-)

Sintaxis

`a - b`

Realiza una operación de resta. Este operador se utiliza con tipos numéricos y devuelve la diferencia del segundo operando restado del primero.

Ejemplo

```
a = 10
b = 4
c = a - b # resta
print(c) # Salidas: 6
```

2322 Operador de multiplicación (*)

Sintaxis

`a * b`

Realiza una operación de multiplicación. Si los operandos son números, devuelve su producto. Este operador también se puede usar con secuencias (como cadenas o listas) para repetirlas.

Ejemplo

```
# Con números
a = 3
b = 4
c = a * b # multiplicación
print(c) # Salidas: 12
```



```
# Con cadenas
text = "abc"
result = text * 2 # repetición
print(result) # Salidas: 'abcabc'
```

2323 Operador de división (/)

ID 2323 · título recuperado de la página española

Sintaxis

a / b

Realiza una operación de división. Devuelve el resultado de dividir el primer operando por el segundo como un número de punto flotante. Este operador siempre produce un resultado de punto flotante, incluso si ambos operandos son enteros.

Ejemplo

```
a = 10
b = 4
c = a / b # división
print(c) # Salidas: 2.5
```

2324 Operador de división entera (//)

ID 2324 · título recuperado de la página española

Sintaxis

a // b

Realiza una división de piso. Divide el primer operando por el segundo, truncando el resultado al entero menor más cercano. El resultado es un entero si ambos operandos son enteros, de lo contrario, es un flotante.

Ejemplo

```
a = 10
b = 4
c = a // b # división de piso
print(c) # Salidas: 2
```

2325 Operador de módulo (%)

ID 2325 · título recuperado de la página española

Sintaxis

a % b

Devuelve el resto de dividir el primer operando por el segundo. Este operador se utiliza a menudo para determinar si un número es par, impar o divisible por otro número.

Ejemplo

```
a = 10
b = 4
c = a % b # módulo
print(c) # Salida: 2
```

2326 Operador de exponenciación (**)

ID 2326 · título recuperado de la página española

Sintaxis

a ** b

Eleva el primer operando a la potencia del segundo. Este operador funciona con números enteros y de punto flotante, y el resultado es un flotante si cualquiera de los operandos es un flotante.

Ejemplo

```
a = 2
b = 3
c = a ** b # exponenciación
print(c) # Salidas: 8
```

1936 Operadores de comparación

ID 1936 · título localizado en HelpEs

Los operadores de comparación se utilizan para comparar dos valores o expresiones. Devuelven un valor booleano (True o False) basado en la relación entre los operandos. Estos operadores incluyen comprobaciones de igualdad, desigualdad, mayor que, menor que y sus combinaciones.

2327 Operador de igualdad (==)

ID 2327 · título recuperado de la página española

Sintaxis

```
a == b
```

Verifica si los dos operandos son iguales. Devuelve True si los valores de ambos operandos son iguales, de lo contrario devuelve False.

Ejemplo

```
a = 5
b = 5
result = a == b # verificación de igualdad
print(result) # Salida: True
```

2328 Operador de desigualdad (!=)

ID 2328 · título recuperado de la página española

Sintaxis

```
a != b
```

Verifica si los dos operandos no son iguales. Devuelve True si los valores de los operandos son diferentes, de lo contrario devuelve False.

Ejemplo

```
a = 5
b = 3
result = a != b # verificación de desigualdad
print(result) # Salidas: True
```

2329 Operador mayor que (>)

ID 2329 · título recuperado de la página española

Sintaxis

```
a > b
```

Comprueba si el primer operando es mayor que el segundo. Devuelve True si el primer operando es mayor, de lo contrario devuelve False.

Ejemplo

```
a = 5
b = 3
result = a > b # verificación de mayor que
print(result) # Salidas: True
```

2330 Operador menor que (<)

ID 2330 · título recuperado de la página española

Sintaxis

`a < b`

Comprueba si el primer operando es menor que el segundo. Devuelve True si el primer operando es menor, de lo contrario devuelve False.

Ejemplo

```
a = 3
b = 5
result = a < b # verificación de menor que
print(result) # Salidas: True
```

2331 Operador mayor o igual (>=)

ID 2331 · título recuperado de la página española

Sintaxis

`a >= b`

Verifica si el primer operando es mayor o igual que el segundo. Devuelve True si el primer operando es mayor o igual, de lo contrario devuelve False.

Ejemplo

```
a = 5
b = 5
result = a >= b # verificación mayor o igual
print(result) # Salidas: True
```

2332 Operador menor o igual (<=)

ID 2332 · título recuperado de la página española

Sintaxis

`a <= b`

Comprueba si el primer operando es menor o igual que el segundo. Devuelve True si el primer operando es menor o igual, de lo contrario devuelve False.

Ejemplo

```
a = 3
b = 5
result = a <= b # verificación menor o igual
print(result) # Salidas: True
```

1937 Operadores a nivel de bits

ID 1937 · título localizado en HelpEs

Los operadores a nivel de bits realizan operaciones directamente sobre la representación binaria de los números. Se utilizan comúnmente para tareas de programación como establecer, borrar o alternar bits específicos en un número.

2335 Operador AND-bit (&)

ID 2335 · título recuperado de la página española

Sintaxis

`a & b`

Realiza una operación AND a nivel de bits en dos enteros. Este operador compara cada bit de sus operandos y devuelve un entero con los bits establecidos en 1 donde los bits correspondientes de ambos operandos son 1.

Ejemplo

```
a = 12 # en binario: 1100
b = 10 # en binario: 1010
c = a & b # operación AND a nivel de bits
print(bin(c)) # Salida: '0b1000'
```

2336 Operador OR-bit (|)

ID 2336 · título recuperado de la página española

Sintaxis

`a | b`

Realiza una operación OR bit a bit inclusiva entre dos enteros. Este operador compara cada bit de sus operandos y devuelve un nuevo entero con bits establecidos en 1 si al menos uno de los bits correspondientes de los operandos es 1.

Ejemplo

```
a = 12 # en binario: 1100
b = 10 # en binario: 1010
c = a | b # operación OR a nivel de bits
print(bin(c)) # Salida: '0b1110'
```

2337 Operador XOR-bit (^)

ID 2337 · título recuperado de la página española

Sintaxis

`a ^ b`

Realiza una operación XOR a nivel de bits entre dos enteros. Este operador compara cada bit de sus operandos y devuelve un nuevo entero con bits establecidos en 1 donde los bits correspondientes de los operandos son diferentes, y 0 donde son iguales.

Ejemplo

```
a = 12 # en binario: 1100
b = 10 # en binario: 1010
c = a ^ b # operación XOR a nivel de bits
print(bin(c)) # Salida: '0b0110'
```

2338 Operador NOT-bit (~)

ID 2338 · título recuperado de la página española

Sintaxis

`~ a`

El operador NOT a nivel de bits (~) es un operador unario que realiza una operación de complemento a nivel de bits. Invierte cada bit de su operando, cambiando los bits que están establecidos en 1 a 0, y los bits que están establecidos en 0 a 1.

Ejemplo

```
a = 12 # en binario: 1100
b = ~a # operación NOT a nivel de bits
```

2333 Operador despl-izquierda (<<)

ID 2333 · título recuperado de la página española

Sintaxis

`i << b`

Devuelve un número con los bits del operando `i` desplazados a la izquierda por `b` posiciones. Ambos operandos deben ser enteros.

Ejemplo

```
a = 5 # 101 en binario
b = a << 2 # Desplaza a la izquierda 2 bits
print(bin(b)) # Salida: '0b10100'
```

2334 Operador despl-derecha (>>)

ID 2334 · título recuperado de la página española

Sintaxis

```
i >> b
```

Devuelve un número con los bits del operando *i* desplazados a la derecha por *b* posiciones. Ambos operandos deben ser enteros.

Ejemplo

```
a = 20 # 10100 en binario
b = a >> 2 # Desplaza a la derecha 2 bits
print(bin(b)) # Salida: '0b101'
```

1938 Operadores de asignación

ID 1938 · título localizado en HelpEs

Los operadores de asignación se utilizan para asignar valores a variables. Además del operador de asignación básico, Python proporciona operadores combinados para operaciones aritméticas, bit a bit y otras. Estos operadores simplifican las expresiones al realizar una operación y una asignación en un solo paso.

2339 Operador de asignación

ID 2339 · título recuperado de la página española

Sintaxis

```
a = b
```

Asigna el valor del operando derecho al operando izquierdo. Este operador se utiliza para inicializar o actualizar el valor de una variable.

Ejemplo

```
a = 5 # Asigna el valor 5 a la variable a
print(a) # Imprime: 5
```

2340 Operador suma-asignación

ID 2340 · título recuperado de la página española

Sintaxis

```
a += b
```

Suma el valor del operando derecho al operando izquierdo y asigna el resultado al operando izquierdo. Este operador funciona no solo con números, sino también con otros tipos de datos como cadenas, tuplas y listas, donde realiza concatenación o extensión.

Ejemplo

```
a = 5
a += 3 # Equivalente a `a = a + 3`
print(a) # Imprime: 8
```

2341 Operador resta-asignación

ID 2341 · título recuperado de la página española

Sintaxis

```
a -= b
```

Resta el valor del operando derecho del operando izquierdo y asigna el resultado al operando izquierdo.

Ejemplo

```
a = 10
a -= 3 # Equivalente a `a = a - 3`
print(a) # Imprime: 7
```

2342 Operador multiplicación-asignación

ID 2342 · título recuperado de la página española

Sintaxis

```
a *= b
```

Multiplica el operando de la izquierda por el operando de la derecha y asigna el resultado al operando de la izquierda. Este operador funciona no solo con números, sino también con cadenas, tuplas y listas, donde realiza repetición.

Ejemplo

```
a = 4
a *= 2 # Equivalente a `a = a * 2`
print(a) # Imprime: 8
```

2343 Operador división-asignación

ID 2343 · título recuperado de la página española

Sintaxis

```
a /= b
```

Divide el operando de la izquierda por el operando de la derecha y asigna el resultado al operando de la izquierda.

Ejemplo

```
a = 10
a /= 2 # Equivalente a `a = a / 2`
print(a) # Imprime: 5.0
```

2344 Operador división-piso-asignación

ID 2344 · título recuperado de la página española

Sintaxis

```
a //= b
```

Realiza una división entera entre el operando izquierdo y el operando derecho, y asigna el resultado al operando izquierdo.

Ejemplo

```
a = 10
a //= 3 # Equivalente a `a = a // 3`
print(a) # Imprime: 3
```

2345 Operador módulo-asignación

ID 2345 · título recuperado de la página española

Sintaxis

```
a %= b
```

Encuentra el resto de dividir el operando izquierdo por el operando derecho y asigna el resultado al operando izquierdo.

Ejemplo

```
a = 10
a %= 3 # Equivalente a `a = a % 3`
print(a) # Imprime: 1
```

2346 Operador exponenciación-asignación

ID 2346 · título recuperado de la página española

Sintaxis

```
a **= b
```

Eleva el operando izquierdo a la potencia del operando derecho y asigna el resultado al operando izquierdo.

Ejemplo

```
a = 2
a **= 3 # Equivalente a `a = a ** 3`
print(a) # Imprime: 8
```

2347 Operador AND-bit-asignación

ID 2347 · título recuperado de la página española

Sintaxis

```
a &= b
```

Realiza una operación AND a nivel de bits entre el operando izquierdo y el operando derecho y asigna el resultado al operando izquierdo.

Ejemplo

```
a = 12 # en binario: 1100
b = 10 # en binario: 1010
a &= b # Equivalente a `a = a & b`
print(a) # Imprime: 8 (binario: 1000)
```

2348 Operador OR-bit-asignación

ID 2348 · título recuperado de la página española

Sintaxis

```
a |= b
```

Realiza una operación OR a nivel de bits entre el operando izquierdo y el operando derecho y asigna el resultado al operando izquierdo.

Ejemplo

```
a = 12 # en binario: 1100
b = 10 # en binario: 1010
a |= b # Equivalente a `a = a | b`
print(a) # Imprime: 14 (binario: 1110)
```

2349 Operador XOR-bit-asignación

ID 2349 · título recuperado de la página española

Sintaxis

```
a ^= b
```

Realiza una operación XOR a nivel de bits entre los operandos izquierdo y derecho y asigna el resultado al operando izquierdo.

Ejemplo

```
a = 12 # en binario: 1100
b = 10 # en binario: 1010
a ^= b # Equivalente a `a = a ^ b`
print(a) # Imprime: 6 (binario: 0110)
```

2350 Operador despl-izquierda-asignación

ID 2350 · título recuperado de la página española

Sintaxis

```
a <<= b
```

Realiza un desplazamiento a la izquierda a nivel de bits en el operando izquierdo por el número de bits especificado por el operando derecho, y asigna el resultado al operando izquierdo.

Ejemplo

```
a = 5 # en binario: 101
a <<= 2 # Equivalente a `a = a << 2`
print(a) # Imprime: 20 (binario: 10100)
```

2351 Operador despl-derecha-asignación

ID 2351 · título recuperado de la página española

Sintaxis

```
a >>= b
```

Realiza un desplazamiento a la derecha a nivel de bits en el operando izquierdo por el número de bits especificado por el operando derecho, y asigna el resultado al operando izquierdo.

Ejemplo

```
a = 20 # en binario: 10100
a >>= 2 # Equivalente a `a = a >> 2`
print(a) # Imprime: 5 (binario: 101)
```

1939 %-operador de formateo

ID 1939 · título localizado en HelpEs

Sintaxis

```
"format string" % value
"format string" % (value1, value2, ...)
"format string" % {"key1": value1, ...}
```

Realiza una operación de formateo de cadenas con estilo de porcentaje. La cadena de formato situada a la izquierda del operador % puede contener texto literal y campos de reemplazo que comienzan con un signo de porcentaje. Para incluir un signo de porcentaje literal en el resultado, utilice %%.

Para un único campo de reemplazo, el valor puede proporcionarse directamente. Los valores posicionales múltiples deben proporcionarse como una tupla. Para formatear una tupla como un único valor, esta debe incluirse dentro de una tupla de un solo elemento.

El formateo con nombre utiliza un diccionario. Los campos de reemplazo con nombre deben contener la clave del diccionario entre paréntesis, por ejemplo %(name)s o %(value)d.

Devuelve una nueva cadena en la que cada campo de reemplazo se sustituye por el valor formateado del argumento correspondiente.

Sintaxis de la especificación de formato:

%%[(key)][flags][width][.precision]type

Especificadores de tipo comunes [type]:

- %s: representación de cadena mediante str()
- %r: representación mediante repr()
- %d: entero decimal
- %i: entero decimal; igual que %d
- %u: se acepta y se comporta como %d, también con valores negativos
- %o: entero octal
- %x: entero hexadecimal con letras minúsculas
- %X: entero hexadecimal con letras mayúsculas
- %f: notación de punto fijo; la precisión predeterminada es de 6 decimales
- %F: igual que %f, pero muestra INF y NAN en mayúsculas
- %e: notación exponencial con una e minúscula
- %E: notación exponencial con una E mayúscula
- %g: formato general que utiliza notación de punto fijo o exponencial, según el valor y la precisión
- %G: igual que %g, pero utiliza letras mayúsculas cuando corresponde
- %c: carácter
- %%: signo de porcentaje literal; no consume ningún argumento

En esta implementación, %c acepta un entero o una cadena formada por un único carácter ASCII. Los valores enteros se reducen a un valor de carácter de 8 bits. Los especificadores de tipo %b, %a y %n no son compatibles.

Banderas [flags]:

- -: alinear a la izquierda dentro del ancho de campo disponible

- `+`: mostrar siempre el signo
- (espacio): mostrar un espacio inicial para números positivos y un signo menos para números negativos
- `#`: activar la forma alternativa
- `0`: rellenar los valores numéricos con ceros

Para `%o`, `%x` y `%X`, la forma alternativa añade el prefijo `0o`, `0x` o `0X`, respectivamente. En los formatos de punto flotante, `#` no fuerza la inclusión de un punto decimal ni conserva los ceros finales en esta implementación.

Ancho [width]:

- width: ancho mínimo del campo; los valores se alinean a la derecha de forma predeterminada
- `*`: obtener el ancho del campo del siguiente argumento posicional

Se admite un ancho dinámico positivo. En esta implementación, un ancho dinámico negativo no produce el comportamiento estándar de alineación a la izquierda y puede hacer que no se aplique ningún relleno.

Precisión [.precision]:

- .precision: especificar directamente la precisión
- `*`: obtener la precisión del siguiente argumento posicional

Para `%f`, `%F`, `%e` y `%E`, la precisión especifica el número de dígitos después del punto decimal. Para `%g` y `%G`, especifica el número de cifras significativas. Para las cadenas, especifica el número máximo de caracteres. Para los enteros, especifica el número mínimo de dígitos y añade ceros iniciales cuando es necesario.

Ejemplos

```
"¡Hola %s!" % "mundo"
# '¡Hola mundo!'
```

```
"Número: %d" % 42
# 'Número: 42'
```

```
"Flotante: %.2f" % 3.14159
# 'Flotante: 3.14'
```

```
"Nombre: %s, Edad: %d" % ("Juan", 25)
# 'Nombre: Juan, Edad: 25'
```

```
"%(nombre)s tiene %(edad)d años" % {"nombre": "Juan", "edad": 25}
# 'Juan tiene 25 años'
```

```
"Rellenado: %05d" % 42
# 'Rellenado: 00042'
```

```
"Derecha: %10s" % "test"
# 'Derecha: test'
```

1940 Excepciones

ID 1940 · título localizado en HelpEs

Las excepciones son eventos que interrumpen el flujo normal de la ejecución del programa, indicando que ha ocurrido un error o una condición inesperada. Proporcionan un mecanismo para manejar errores de manera elegante e implementar una lógica de manejo de errores robusta. Comprender y usar eficazmente las excepciones es crucial para desarrollar aplicaciones estables y eficientes. Las excepciones pueden manejarse utilizando bloques `try` y `except`, lo que permite a los desarrolladores capturar y gestionar errores de manera efectiva. Además, las excepciones pueden generarse programáticamente para imponer ciertas condiciones o manejar casos de maneras específicas. Por ejemplo, en Python, puede generar una excepción utilizando la palabra clave `raise`:

```
def check_age(age):
    if age < 18:
        raise ValueError("¡Acceso denegado!")
    else:
        print("Acceso concedido.")
# Ejemplo de invocación de la función
try:
    check_age(17)
except ValueError as e:
    print(e)
```

En este ejemplo, se genera un `ValueError` si la edad de entrada es menor de 18, lo que luego se captura en el bloque `try-except`, permitiendo un manejo elegante de esta condición de error.

1941 BaseException, clase

ID 1941 · título localizado en HelpEs

BaseException es la clase base para todas las excepciones integradas en Python. Es la parte superior de la jerarquía de excepciones y está destinada a ser la clase base para excepciones definidas por el usuario. Todas las demás excepciones se derivan de BaseException.

1942 ArithmeticError, clase

ID 1942 · título localizado en HelpEs

ArithmeticError es una clase de excepción incorporada que sirve como clase base para todos los errores relacionados con las operaciones aritméticas. Es una subclase de la clase Exception y se utiliza para capturar varias excepciones relacionadas con la aritmética, como la división por cero, el desbordamiento y los errores de punto flotante.

1943 AssertionError, clase

ID 1943 · título localizado en HelpEs

AssertionError es una clase de excepción incorporada que se genera cuando una declaración assert falla. Indica que una afirmación, que es una ayuda para la depuración que prueba una condición, se ha evaluado como False.

1944 AttributeError, clase

ID 1944 · título localizado en HelpEs

AttributeError es una clase de excepción incorporada que se genera cuando falla una referencia o asignación de atributo. Esto ocurre típicamente cuando un atributo no se encuentra en un objeto.

1945 EOFError, clase

ID 1945 · título localizado en HelpEs

EOFError es una clase de excepción incorporada que se genera cuando la función input() alcanza una condición de fin de archivo (EOF) sin leer ningún dato. Esta excepción ocurre típicamente al leer entradas desde la entrada estándar (como la consola) y no hay más entradas disponibles.

1946 Exception, clase

ID 1946 · título localizado en HelpEs

La excepción es una clase base incorporada para todas las excepciones estándar en Python. Es la categoría de excepciones más general de la que deberían heredar las excepciones definidas por el usuario. Capturar una excepción manejará la mayoría de los errores que ocurren durante la ejecución.

1947 GeneratorExit, clase

ID 1947 · título localizado en HelpEs

GeneratorExit es una clase de excepción incorporada que se genera cuando se llama al método close() de un generador. Señala que el generador se está cerrando y se utiliza internamente por el generador para limpiar recursos.

1948 ImportError, clase

ID 1948 · título localizado en HelpEs

ImportError es una clase de excepción incorporada que se genera cuando una declaración de importación no puede encontrar un módulo o cuando un módulo no se puede cargar. Indica problemas con la importación de módulos, como archivos faltantes o dependencias no resueltas.

1949 IndentationError, clase

ID 1949 · título localizado en HelpEs

IndentationError es una clase de excepción incorporada que se genera cuando hay un problema con la indentación del código. Generalmente ocurre cuando el código no está alineado correctamente de acuerdo con las reglas de

indentación de Python.

1950 IndexError, clase

ID 1950 · título localizado en HelpEs

`IndexError` es una clase de excepción incorporada que se genera cuando un subíndice de secuencia está fuera de rango. Ocurre al intentar acceder a un índice que está fuera de los límites de una lista, tupla u otros tipos de secuencias.

1951 KeyboardInterrupt, clase

ID 1951 · título localizado en HelpEs

`KeyboardInterrupt` es una clase de excepción incorporada en Python que se genera cuando el usuario interrumpe la ejecución del programa, típicamente al presionar [On] en el teclado. Se utiliza para manejar los casos en los que el usuario desea detener el programa en ejecución.

1952 KeyError, clase

ID 1952 · título localizado en HelpEs

`KeyError` es una clase de excepción incorporada que se genera cuando no se encuentra una clave en un diccionario. Ocurre al intentar acceder o modificar un diccionario con una clave que no existe en el diccionario.

1953 LookupError, clase

ID 1953 · título localizado en HelpEs

`LookupError` es una clase de excepción incorporada que sirve como clase base para las excepciones que se generan cuando una operación de búsqueda falla. Es la superclase de `IndexError` y `KeyError`, y se utiliza para errores relacionados con el acceso a elementos en secuencias o mapeos.

1954 MemoryError, clase

ID 1954 · título localizado en HelpEs

`MemoryError` es una clase de excepción incorporada que se genera cuando una operación se queda sin memoria. Ocurre cuando el sistema no puede asignar suficiente memoria para completar la operación, generalmente debido a un uso excesivo de memoria o límites de recursos.

1955 NameError, clase

ID 1955 · título localizado en HelpEs

`NameError` es una clase de excepción incorporada que se genera cuando no se encuentra un nombre local o global. Ocurre cuando se hace referencia a una variable o función antes de que se haya definido o si hay un error tipográfico en el nombre.

1956 NotImplementedError, clase

ID 1956 · título localizado en HelpEs

`NotImplementedError` es una clase de excepción incorporada que señala que un método u operación no ha sido implementado. Se utiliza comúnmente en métodos abstractos o funciones de marcador de posición para indicar que la funcionalidad necesita ser definida en subclases o completada en el desarrollo futuro.

1957 OSError, clase

ID 1957 · título localizado en HelpEs

`OSError` es una clase de excepción incorporada que se genera cuando una operación relacionada con el sistema falla. Esto incluye problemas relacionados con operaciones de archivos, llamadas al sistema o gestión de recursos. Los escenarios comunes en los que puede generarse `OSError` incluyen errores de archivo no encontrado, errores de permisos o problemas relacionados con el hardware. Es una clase base para otras excepciones específicas relacionadas con el sistema operativo.

1958 OverflowError, clase

ID 1958 · título localizado en HelpEs

`OverflowError` es una clase de excepción incorporada que se genera cuando una operación aritmética excede el límite máximo para un tipo numérico. Este error ocurre típicamente con operaciones que involucran enteros o números de punto flotante que resultan en valores demasiado grandes para ser representados dentro de los límites del tipo de datos. Por ejemplo, intentar calcular un número demasiado grande para un entero o exceder la precisión de punto flotante puede desencadenar este error.

1959 `RuntimeError`, clase

ID 1959 · título localizado en HelpEs

`RuntimeError` es una clase de excepción incorporada que se genera cuando se detecta un error que no se encuentra en ninguna otra categoría de excepción específica. Es un error de propósito general utilizado para indicar problemas que ocurren durante la ejecución del programa, como problemas con la lógica del programa o condiciones inesperadas que no están cubiertas por excepciones más específicas.

1960 `StopAsyncIteration`, clase

ID 1960 · título localizado en HelpEs

`StopAsyncIteration` es una clase de excepción incorporada que se genera para señalar el final de una iteración asíncrona. Se utiliza dentro de los iteradores asíncronos para indicar que no hay más elementos para producir, finalizando efectivamente la iteración. Esta excepción es manejada típicamente por el protocolo de iteración asíncrona para terminar los bucles que iteran sobre objetos iterables asíncronos.

1961 `StopIteration`, clase

ID 1961 · título localizado en HelpEs

`StopIteration` es una clase de excepción incorporada que se lanza para señalar el final de una iteración. Se utiliza dentro de los iteradores para indicar que no hay más elementos que devolver. Cuando se lanza esta excepción, señala que la iteración debe detenerse, y generalmente es manejada por el protocolo de iteración para terminar los bucles que iteran sobre objetos iterables.

1962 `SyntaxError`, clase

ID 1962 · título localizado en HelpEs

`SyntaxError` es una clase de excepción incorporada que se genera cuando el analizador encuentra una sintaxis no válida. Indica que hay un problema con la estructura del código, como la falta de puntuación o paréntesis desbalanceados, lo que impide que el código se analice correctamente.

1963 `SystemExit`, clase

ID 1963 · título localizado en HelpEs

`SystemExit` es una clase de excepción incorporada que es levantada por la función `sys.exit()` para salir de un programa. Proporciona una forma de salir del programa y opcionalmente pasar un código de estado de salida. Esta excepción es capturada por el intérprete para terminar el programa de manera ordenada y puede ser utilizada para realizar operaciones de limpieza o señalar una terminación exitosa o no exitosa.

1964 `TypeError`, clase

ID 1964 · título localizado en HelpEs

`TypeError` es una clase de excepción incorporada que se genera cuando se aplica una operación o función a un objeto de tipo inapropiado. Ocurre cuando una operación o función recibe un argumento de un tipo que no se espera o no se admite, como sumar una cadena a un entero o llamar a una función con tipos de argumentos incorrectos.

1965 `UnicodeError`, clase

ID 1965 · título localizado en HelpEs

`UnicodeError` es una clase de excepción incorporada que se genera cuando una operación de codificación o decodificación falla debido a problemas con los datos Unicode. Este error abarca varias excepciones específicas, como `UnicodeEncodeError`, `UnicodeDecodeError` y `UnicodeTranslateError`, que se generan cuando hay

problemas al convertir entre Unicode y otras codificaciones, o al manejar datos de texto no ASCII.

1966 ValueError, clase

ID 1966 · título localizado en HelpEs

ValueError es una clase de excepción incorporada que se genera cuando una función recibe un argumento del tipo correcto pero de un valor inapropiado. Este error ocurre cuando el valor de un argumento no es adecuado para la operación o función, como pasar una cadena que no se puede convertir en un número o proporcionar un valor fuera de rango a una función.

1967 ZeroDivisionError, clase

ID 1967 · título localizado en HelpEs

ZeroDivisionError es una clase de excepción incorporada que se genera cuando se realiza una operación de división o módulo con cero como divisor. Este error indica que se intentó dividir un número por cero, lo cual es matemáticamente indefinido.

1754 PYTHON

ID 1754 · título localizado en HelpEs

Sintaxis

```
PYTHON("script" [, Argv0, Argv1, ..., ArgvN])
```

Cuando se utiliza como función en un programa HP PPL, PYTHON acepta uno o más argumentos.

El primer argumento es el script de Python que se ejecutará, especificado como una cadena de caracteres. Esta cadena puede introducirse directamente o almacenarse en una variable.

Los demás argumentos pasados a la función PYTHON se convierten en cadenas de caracteres (si no lo son ya) y se almacenan en la variable argv de Python, donde se puede acceder a ellos de la misma manera que a los argumentos de línea de comandos en un script de Python ejecutado en un PC.

La palabra clave PYTHON también puede utilizarse de otras dos maneras en HP PPL.

La siguiente estructura, situada en el nivel superior de un programa HP PPL:

```
#PYTHON name
Script de Python
#end
```

crea implícitamente una variable global de tipo cadena de caracteres llamada "name", que puede utilizarse como cualquier otra variable global del programa (leerse, recibir valores, utilizarse, etc.).

El uso previsto consiste en pasar esta variable como primer argumento en una llamada a la función PYTHON.

Ejemplo

```
#PYTHON test
import sys
print(str(1+2)+sys.argv[0])
#end
```

```
EXPORT PPL(a)
BEGIN
PYTHON(test, a);
END;
```

PPL(5) imprime "35".

La última forma de utilizar PYTHON es mediante la siguiente construcción:

```
#PYTHON test2(a)
import sys
print("el parámetro es "+sys.argv[0])
#end
```

```
EXPORT PPL2(a)
BEGIN
test2(a);
END;
```

Una vez dentro del sistema Python, es posible interactuar con el sistema PPL mediante la función `hpprime.eval("«código PPL»")`, que permite enviar comandos al sistema PPL y obtener un resultado. La cadena puede, por supuesto, construirse dinámicamente para realizar prácticamente cualquier operación deseada.

0805 Índice de comandos de Inicio

ID 805 · título localizado en HelpEs

Lista de todas las funciones y comandos disponibles en la vista de inicio (PPL).

0806 Funciones en la vista de Inicio (PPL): 'A'

ID 806 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'A'

0628 Valor de absoluto

ID 628 · título recuperado de la página española

Sintaxis

$\text{ABS}(\text{expr})$
 $\text{ABS}(\text{matriz})$

Para argumentos numéricos, devuelve el valor absoluto de la expresión.

Para argumentos de matriz, devuelve la norma Frobenius (euclidiano) de la matriz.

0886 Coseno inverso

ID 886 · título recuperado de la página española

Sintaxis

$\text{ACOS}(\text{Valor})$

Devuelve el coseno inverso de Valor.

El resultado depende de la configuración de Medida del ángulo en Configuración de Inicio, Configuración CAS o Configuración Simbólica.

Ejemplos

[Ejemplo gráfico E1] (en modo grados)
[Ejemplo gráfico E2]
[Ejemplo gráfico E3] (en modo grados)

0564 Coseno hiperbólico inverso

ID 564 · título recuperado de la página española

Sintaxis

$\text{ACOSH}(\text{valor})$

Coseno hiperbólico inverso: $\text{COSH}^{-1}(X)$

0559 Arco cotangente

ID 559 · título recuperado de la página española

Sintaxis

$\text{ACOT}(\text{valor})$

Cotangente inverso: $\text{COT}^{-1}(X)$

Ejemplos

[Ejemplo gráfico E1] (en modo grados)
[Ejemplo gráfico E2]
[Ejemplo gráfico E3] (en modo grados)

0555 Arco cosecante

ID 555 · título recuperado de la página española

Sintaxis

ACSC(valor)

Cosecante inverso: $\text{CSC}^{-1}(X)$

Ejemplos

[Ejemplo gráfico E1] (en modo grados)

[Ejemplo gráfico E2]

[Ejemplo gráfico E3] (en modo grados)

0887 Agregar columna

ID 887 · título recuperado de la página española

Sintaxis

ADDCOL(matrix, vector, column_number)

Inserta valores del vector vector en una columna antes de column_number en la matriz especificada matrix.

El tamaño de vector debe ser igual al número de filas en la matrix.

La matriz se modifica directamente. No es necesario asignar de nuevo el resultado a la variable.

0891 Agregar fila

ID 891 · título recuperado de la página española

Sintaxis

ADDFROW(matrix, vector, row_number)

Inserta valores del vector vector en una fila antes de row_number en la matriz matrix especificada.

El tamaño de vector debe ser igual al número de columnas en la matrix.

La matriz se modifica directamente. No es necesario asignar de nuevo el resultado a la variable.

0068 Archivos de aplicación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Archivos de aplicación.](#)

0069 Archivos de aplicación binaria

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Archivos de aplicación binaria.](#)

0550 Antilogaritmo común

ID 550 · título recuperado de la página española

Sintaxis

ALOG(valor)

Exponencial común: 10^x (antilogaritmo)

Devuelve el resultado de elevar 10 a la potencia de valor.

0895 AND lógico

ID 895 · título recuperado de la página española

Sintaxis

Valor1 AND Valor2

Para números reales, devuelve 1 si Valor1 y Valor2 no son un cero; de lo contrario, devuelve 0.
Para enteros y cadenas, AND se realiza a nivel de bit, y devuelve 1 si los bits correspondientes son 1, de lo contrario, devuelve 0.

0066 Lista de aplicaciones

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Lista de aplicaciones.](#)

0901 Dibujar arco

ID 901 · título recuperado de la página española

Sintaxis

ARC(G, X, Y, Radius [, $\angle 1$, $\angle 2$] [, Color])

Dibuja un círculo, una elipse o un arco en el GROB G mediante coordenadas cartesianas.

Cuando Radius es un único valor real, la función dibuja un círculo. Cuando Radius es una lista de dos elementos, dibuja una elipse. Si se especifican $\angle 1$ y $\angle 2$, solo se dibuja el arco correspondiente.

Parámetros

- G: GROB de destino.
- X, Y: Coordenadas cartesianas del centro.
- Radius: Radio del círculo o una lista con la forma {Radius_X, Radius_Y} que especifica los radios horizontal y vertical de una elipse.
- $\angle 1$, $\angle 2$: Ángulos opcionales inicial y final del arco, interpretados según el modo angular actual.
- Color: Especificación de color opcional. Un único color especifica el color del borde. Una lista con la forma {Border_color [, Fill_color]} especifica el color del borde y, opcionalmente, el color de relleno.

Cuando se omiten $\angle 1$ y $\angle 2$, ARC dibuja un círculo o una elipse completos.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

0902 Dibujar arco

ID 902 · título recuperado de la página española

Sintaxis

ARC_P(G, X, Y, Radius [, $\angle 1$, $\angle 2$] [, Color])

Dibuja un círculo, una elipse o un arco en el GROB G mediante coordenadas de píxel.

Cuando Radius es un único valor real, la función dibuja un círculo. Cuando Radius es una lista de dos elementos, dibuja una elipse. Si se especifican $\angle 1$ y $\angle 2$, solo se dibuja el arco correspondiente.

Parámetros

- G: GROB de destino.
- X, Y: coordenadas de píxel del centro.
- Radius: Radio del círculo o una lista con la forma {Radius_X, Radius_Y} que especifica los radios horizontal y vertical de una elipse.
- $\angle 1$, $\angle 2$: Ángulos opcionales inicial y final del arco, interpretados según el modo angular actual.
- Color: Especificación de color opcional. Un único color especifica el color del borde. Una lista con la forma {Border_color [, Fill_color]} especifica el color del borde y, opcionalmente, el color de relleno.

Cuando se omiten $\angle 1$ y $\angle 2$, ARC_P dibuja un círculo o una elipse completos.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

0544 Argumento

ID 544 · título recuperado de la página española

Sintaxis
 $\text{ARG}(x+yi)$

Encuentra el ángulo determinado por un número complejo.

Ejemplos

[Ejemplo gráfico E1] (modo grados)
[Ejemplo gráfico E2]

0905 ASC — Devuelve una lista que contiene los valores Unicode numéricos de la cadena.

ID 905 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ASC

Explicación española: Devuelve una lista que contiene los valores Unicode numéricos de la cadena.

Sintaxis
 $\text{ASC}(\text{Cadena})$

Devuelve una lista que contiene los valores Unicode numéricos de la cadena.

0557 Arco secante

ID 557 · título recuperado de la página española

Sintaxis
 $\text{ASEC}(\text{valor})$

Secante inverso: $\text{SEC}^{-1}(X)$

Ejemplos

[Ejemplo gráfico E1] (en modo grados)
[Ejemplo gráfico E2]
[Ejemplo gráfico E3] (en modo grados)

0906 Seno inverso

ID 906 · título recuperado de la página española

Sintaxis
 $\text{ASIN}(\text{Valor})$

Devuelve el seno inverso de Valor.

El resultado depende de la configuración de Medida del ángulo en Configuración de Inicio, Configuración CAS o Configuración Simbólica.

Ejemplos

[Ejemplo gráfico E1] (en modo grados)
[Ejemplo gráfico E2]
[Ejemplo gráfico E3] (en modo grados)

0562 Seno hiperbólico inverso

ID 562 · título recuperado de la página española

Sintaxis
 $\text{ASINH}(\text{valor})$

Seno hiperbólico inverso: $\text{SINH}^{-1}(X)$

0908 Tangente inversa

ID 908 · título recuperado de la página española

Sintaxis

ATAN(Valor)

Devuelve la tangente inversa de Valor.

El resultado depende de la configuración de Medida del ángulo en Configuración de Inicio, Configuración CAS o Configuración Simbólica.

Ejemplos

[Ejemplo gráfico E1] (en modo grados)

[Ejemplo gráfico E2]

[Ejemplo gráfico E3] (en modo grados)

0566 Tangente hiperbólica inversa

ID 566 · título recuperado de la página española

Sintaxis

ATANH(valor)

Tangente hiperbólica inversa: $\text{TANH}^{-1}(X)$

0807 Funciones en la vista de Inicio (PPL): 'B'

ID 807 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'B'

0910 Bloque BEGIN END

ID 910 · título recuperado de la página española

Sintaxis

BEGIN comandos; END;

Define un conjunto de comandos a ser ejecutados en un bloque.

Ejemplo

EXPORT SQM1(X)

BEGIN

RETURN X^2-1 ;

END;

Este programa define una función de usuario denominada SQM1(X). Si ingresa SQM1(8) devuelve 63.

0582 Densidad de probabilidad binomial

ID 582 · título recuperado de la página española

Sintaxis

BINOMIAL(n, p, k)

Función de densidad de probabilidad binomial.

Calcula la probabilidad de k éxitos en una secuencia de n ensayos, cada uno con una probabilidad de éxito de p. Tenga en cuenta que n y k son números enteros donde $k \leq n$.

0590 Binomial acumulada

ID 590 · título recuperado de la página española

Sintaxis

BINOMIAL_CDF(n, p, k [, k2])

Función de distribución binomial acumulada

Devuelve la probabilidad de k o menos éxitos de las pruebas n , con una probabilidad de éxito, p para cada prueba. Observe que n y k son enteros con $k \leq n$. Con el cuarto argumento opcional $k2$, devuelve la probabilidad acumulada de que los dos valores de k ; es decir, la probabilidad de éxito entre k y $k2$.

0599 Binomial acumulada inversa

ID 599 · título recuperado de la página española

Sintaxis

`BINOMIAL_ICDF(n, p, q)`

Función de distribución binomial acumulada inversa

Devuelve el número de éxitos k , de una secuencia de n ensayos, cada uno con una probabilidad de p , tal que la probabilidad de k o menos éxitos es q .

0916 AND de bit lógico

ID 916 · título recuperado de la página española

Sintaxis

`BITAND(ent1, ent2, ... entn)`

Devuelve el AND de bit lógico (conjunción) de los números enteros especificados.

0917 NOT de bit lógico

ID 917 · título recuperado de la página española

Sintaxis

`BITNOT(int)`

Devuelve el NOT de bit lógico (negación) del número entero especificado.

0918 OR de bit lógico

ID 918 · título recuperado de la página española

Sintaxis

`BITOR(ent1, ent2, ... entn)`

Devuelve el OR de bit lógico (disyunción) del número entero especificado.

0919 Desplazamiento lógico de bits hacia la izquierda

ID 919 · título recuperado de la página española

Sintaxis

`BITSL(ent1 [, ent2])`

Toma uno o dos enteros como entrada y devuelve el resultado de desplazar los bits del primer entero a la izquierda el número de posiciones indicado por el segundo entero. Si no hay un segundo entero, los bits del primer entero se desplazan a la izquierda una posición.

0920 Desplazamiento lógico de bits hacia la derecha

ID 920 · título recuperado de la página española

Sintaxis

`BITSR(ent1 [, ent2])`

Toma uno o dos enteros como entrada y devuelve el resultado de desplazar los bits del primer entero a la derecha el número de posiciones indicado por el segundo entero. Si no hay un segundo entero, los bits del primer entero se desplazan a la derecha una posición.

0921 XOR de bit lógico

ID 921 · título recuperado de la página española

Sintaxis

```
BITXOR(ent1, ent2, ... entn)
```

Devuelve el XOR de bit lógico (disyunción exclusiva) del número entero especificado.

0922 Copiar GROB

ID 922 · título recuperado de la página española

Sintaxis

```
BLIT([trgtG,] [dx1, dy1 [, dx2, dy2]], srcG [, sx1, sy1 [, sx2, sy2 [, c [, alpha]]]])
```

Copia una región rectangular del GROB srcG en una región rectangular del GROB trgtG mediante coordenadas cartesianas.

Los píxeles de srcG que coincidan con c pueden excluirse de la copia. Los píxeles copiados también pueden dibujarse con una opacidad especificada.

Parámetros

- trgtG: GROB de destino.
- dx1, dy1: Coordenadas cartesianas de la esquina superior izquierda de la región de destino.
- dx2, dy2: Coordenadas cartesianas opcionales de la esquina inferior derecha de la región de destino.
- srcG: GROB de origen.
- sx1, sy1: Coordenadas cartesianas opcionales de la esquina superior izquierda de la región de origen.
- sx2, sy2: Coordenadas cartesianas opcionales de la esquina inferior derecha de la región de origen.
- c: Color de origen opcional que no se copia.
- alpha: Opacidad opcional de los píxeles copiados, de 0 (completamente transparentes) a 255 (completamente opacos).

Regiones de origen y destino

La región de origen se define mediante los puntos (sx1, sy1) y (sx2, sy2). Si se omiten las coordenadas de origen, se utiliza todo el GROB de origen.

La región de destino se define mediante los puntos (dx1, dy1) y (dx2, dy2). Si se omiten dx1 y dy1, se utiliza la esquina superior izquierda de trgtG. Si se omiten dx2 y dy2, se calculan de modo que la región de destino tenga las mismas dimensiones que la región de origen.

Clave de color y opacidad

Si se especifica c, no se copian los píxeles de la región de origen que coincidan con c. Si se omite c, se copian todos los píxeles de la región de origen.

Si se omite alpha, se utiliza una opacidad de 255.

Al especificar c o alpha, proporcione explícitamente las coordenadas de origen para que los argumentos puedan interpretarse sin ambigüedad.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

0923 Copiar GROB

ID 923 · título recuperado de la página española

Sintaxis

```
BLIT_P([trgtG,] [dx1, dy1 [, dx2, dy2]], srcG [, sx1, sy1 [, sx2, sy2 [, c [, alpha]]]])
```

Copia una región rectangular del GROB srcG en una región rectangular del GROB trgtG mediante coordenadas de píxel.

Los píxeles de srcG que coincidan con c pueden excluirse de la copia. Los píxeles copiados también pueden dibujarse con una opacidad especificada.

Parámetros

- `trgtG`: GROB de destino.
- `dx1, dy1`: coordenadas de píxel de la esquina superior izquierda de la región de destino.
- `dx2, dy2`: coordenadas de píxel opcionales de la esquina inferior derecha de la región de destino.
- `srcG`: GROB de origen.
- `sx1, sy1`: coordenadas de píxel opcionales de la esquina superior izquierda de la región de origen.
- `sx2, sy2`: coordenadas de píxel opcionales de la esquina inferior derecha de la región de origen.
- `c`: Color de origen opcional que no se copia.
- `alpha`: Opacidad opcional de los píxeles copiados, de 0 (completamente transparentes) a 255 (completamente opacos).

Regiones de origen y destino

La región de origen se define mediante los puntos (`sx1, sy1`) y (`sx2, sy2`). Si se omiten las coordenadas de origen, se utiliza todo el GROB de origen.

La región de destino se define mediante los puntos (`dx1, dy1`) y (`dx2, dy2`). Si se omiten `dx1` y `dy1`, se utiliza la esquina superior izquierda de `trgtG`. Si se omiten `dx2` y `dy2`, se calculan de modo que la región de destino tenga las mismas dimensiones que la región de origen.

Clave de color y opacidad

Si se especifica `c`, no se copian los píxeles de la región de origen que coincidan con `c`. Si se omite `c`, se copian todos los píxeles de la región de origen.

Si se omite `alpha`, se utiliza una opacidad de 255.

Al especificar `c` o `alpha`, proporcione explícitamente las coordenadas de origen para que los argumentos puedan interpretarse sin ambigüedad.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

0924 `bounded_function` — Devuelve el argumento devuelto por una función de límite, indicando así que la función es limitada.

ID 924 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `bounded_function`

Explicación española: Devuelve el argumento devuelto por una función de límite, indicando así que la función es limitada.

Devuelve el argumento devuelto por una función de límite, indicando así que la función es limitada.

0925 Romper bucle

ID 925 · título recuperado de la página española

Sintaxis

`BREAK [n];`

Sale del bucle actual o de múltiples bucles anidados.

Código de ejemplo:

```
FOR A FROM 1 TO 10 DO
B:=(A + 3) MOD 5;
IF B==1 THEN
BREAK;
END;
END;
```

Si se especifica `n`, `BREAK [n]` sale de `n` niveles de bucles anidados.

0927 Base a real

ID 927 · título recuperado de la página española

Sintaxis

B→R(#entero[m])

Un número entero en base m se convierte en un número entero decimal (base10).

El marcador de base m puede ser b (para binario), o (para octal) o h (para hexadecimal). Si se omite m, se supone la base del sistema actual.

Ejemplos

[Ejemplo gráfico E1]
[Ejemplo gráfico E2] (Si la base del sistema es hexadecimal)
[Ejemplo gráfico E3]

0808 Funciones en la vista de Inicio (PPL): 'C'

ID 808 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'C'

0929 Evaluación CAS

ID 929 · título recuperado de la página española

Sintaxis

CAS(expresión)
CAS.función(...)
CAS.variable[...]

Evalúe una expresión o variable usando el CAS.

Tenga en cuenta que los resultados en modo numérico se transforman en cadenas o listas de expresiones para matrices simbólicas.

0930 CASE — Inicia una estructura de rama "CASE ... END".

ID 930 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CASE
Explicación española: Inicia una estructura de rama "CASE ... END".

Sintaxis

```
CASE
  IF test1 THEN commands1 END
  IF test2 THEN commands2 END
  ...
  IF testN THEN commandsN END
  [DEFAULT] [commandsD]
END;
```

Inicia una estructura de rama "CASE ... END".

Evalúa test1. Si es verdadero, ejecuta commands1 y termina el CASE. De lo contrario, evalúa test2. Si es verdadero, ejecuta commands2. Continúa evaluando pruebas hasta que se encuentre una verdadera. Si no se encuentra ninguna prueba verdadera, ejecuta commandsD, si se proporciona.

0529 Techo

ID 529 · título recuperado de la página española

Sintaxis

CEILING(valor)

Entero más pequeño mayor o igual a valor.

0936 CHAR — Devuelve la cadena correspondiente a los códigos de carácter Unicode numéricos en Lista o Vector, o el...

ID 936 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CHAR

Explicación española: Devuelve la cadena correspondiente a los códigos de carácter Unicode numéricos en Lista o Vector, o el...

Sintaxis

CHAR(Lista)
CHAR(Vector)
CHAR(Entero)

Devuelve la cadena correspondiente a los códigos de carácter Unicode numéricos en Lista o Vector, o el código de carácter Unicode numérico de Entero.

0580 Densidad χ^2

ID 580 · título recuperado de la página española

Sintaxis

CHISQUARE(d, x)

χ^2 (Chi cuadrado) Función de densidad de probabilidad

Calcula la densidad de probabilidad de la distribución χ^2 en x, con d grados de libertad.

0588 Acumulada χ^2

ID 588 · título recuperado de la página española

Sintaxis

CHISQUARE_CDF(d, x [, x2])

Función de distribución χ^2 (Chi cuadrado) acumulada

Con dos valores (d y x) devuelve la probabilidad de cola más baja de la función de densidad de probabilidad χ^2 para el valor x, dados d grados de libertad. Con el tercer argumento opcional x2, devuelve el área debajo de la función de densidad de probabilidad χ^2 entre los dos valores de x.

0596 Acumulada inversa χ^2

ID 596 · título recuperado de la página española

Sintaxis

CHISQUARE_ICDF(d, p)

Inversa de la función de distribución χ^2 (Chi cuadrado) acumulada.

Devuelve el valor de x de forma que la probabilidad de cola inferior χ^2 de x, con d grados de libertad, sea p.

0938 Cuadro de elección

ID 938 · título recuperado de la página española

Sintaxis

CHOOSE(var, "title", "item1", ..., "item14")
CHOOSE(var, "title" {, "item1", ..., "itemN"})

Muestra un cuadro de selección con el "title" dado y que contiene elementos con las cadenas "item1", ..., etc. El elemento correspondiente al valor de var se resalta como la selección predeterminada.

Si el usuario elige un objeto, la variable var se actualiza para contener el número del objeto seleccionado (un entero, 1, 2, 3, ...) y CHOOSE devuelve true (no cero).

Si el usuario sale sin elegir, var no se cambia y CHOOSE devuelve false (0).

1721 CHOOSEDATE — Muestra un selector de fecha de calendario con un título opcional.

ID 1721 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CHOOSEDATE

Explicación española: Muestra un selector de fecha de calendario con un título opcional.

Sintaxis

CHOOSEDATE(var [, title [, min_date [, max_date]])

Muestra un selector de fecha de calendario con un título opcional. El rango de fechas seleccionable puede limitarse especificando la fecha mínima min_date y la fecha máxima max_date. Si se omite alguno de los límites, se permite cualquier fecha válida para el sistema en ese lado del rango.

La fecha seleccionada inicialmente se determina de la siguiente manera: si var contiene una fecha válida dentro del rango permitido, se selecciona esa fecha. De lo contrario, se selecciona la primera fecha válida dentro del rango especificado. Si no se especifica ningún rango, se selecciona la fecha actual del sistema.

El calendario se puede navegar mediante gestos de arrastre. El arrastre horizontal proporciona una navegación rápida mes a mes, mientras que el arrastre vertical proporciona una navegación rápida año a año.

Si el usuario selecciona una fecha, var se actualiza con la fecha seleccionada en el formato YYYY.MMDD, y CHOOSEDATE devuelve verdadero (distinto de cero).

Si el usuario sale sin seleccionar una fecha, var permanece sin cambios y CHOOSEDATE devuelve falso (0).

0630 Norma de columna

ID 630 · título recuperado de la página española

Sintaxis

COLNORM(matriz)

Encuentra el valor máximo (de todas las columnas) de las sumas de los valores absolutos de todos los elementos en una matriz.

0569 Combinaciones

ID 569 · título recuperado de la página española

Sintaxis

COMB(n, r)

Devuelve el número de combinaciones (sin importar el orden) de n cosas teniendo en cuenta r por vez: $n!/(r!(n-r)!)$

0606 Concatenar

ID 606 · título recuperado de la página española

Sintaxis

CONCAT(Object_1 [, Object_2, ..., Object_N])

Toma un conjunto de objetos de entrada y los concatena en una lista. Si un objeto de entrada es otra lista, se aplanan en sus elementos antes de la concatenación. Para evitar que una lista se aplaste, debe estar anidada (una lista dentro de una lista, ver ejemplos).

0633 Número de condición

ID 633 · título recuperado de la página española

Sintaxis

COND(matriz)

Calcula el número de condición de una matriz cuadrada utilizando la norma 1 (norma máxima de suma absoluta por columnas).

0545 Complejo conjugado

ID 545 · título recuperado de la página española

Sintaxis

CONJ(x+yi)

Inversión de signo de la parte imaginaria de un número complejo.

1732 CONST — En un programa, la palabra clave **CONST** puede utilizarse para declarar constantes globales a nivel de...

ID 1732 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CONST

Explicación española: En un programa, la palabra clave **CONST** puede utilizarse para declarar constantes globales a nivel de...

Sintaxis

[EXPORT] CONST Cns1:=Val1 [, Cns2:=Val2, ...];

En un programa, la palabra clave **CONST** puede utilizarse para declarar constantes globales a nivel de programa.

Cualquier intento de asignar un nuevo valor a una constante de este tipo producirá un error en tiempo de ejecución. Esto evita que el usuario modifique accidentalmente valores globales.

Una única instrucción **CONST** puede declarar hasta 8 constantes.

0952 CONTINUE — Transfiere la ejecución en un bucle al principio de la siguiente iteración del enésimo bucle superior.

ID 952 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CONTINUE

Explicación española: Transfiere la ejecución en un bucle al principio de la siguiente iteración del enésimo bucle superior.

Sintaxis

CONTINUE [n];

Transfiere la ejecución en un bucle al principio de la siguiente iteración del enésimo bucle superior. (bucle actual predeterminado).

0953 Convertir

ID 953 · título recuperado de la página española

Sintaxis

CONVERT(Valor_Unidad1, 1_Unidad2)
Valor_Unidad1►_Unidad2

Convierte Valor_Unidad1 al valor correspondiente en Unidad2 compatible.

1737 CoordinateConvert — Convertir entre coordenadas cartesianas, polares, cilíndricas y esféricas.

ID 1737 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: CoordinateConvert

Explicación española: Convertir entre coordenadas cartesianas, polares, cilíndricas y esféricas.

Sintaxis

CoordinateConvert(formatoEntrada, formatoSalida, a, b [, c])
CoordinateConvert(formatoEntrada, formatoSalida, {a, b [, c]})
CoordinateConvert(formatoEntrada, formatoSalida, (a, b))
CoordinateConvert(formatoEntrada, formatoSalida, [matriz])

Convertir entre coordenadas cartesianas, polares, cilíndricas y esféricas.

Los parámetros de formato describen el formato utilizado para la entrada y la salida. Estos pueden ser:

- cartesiano

- polar o cilíndrico polar o cilíndrico con el ángulo marcado para la visualización HMS (este es un formato solo de salida) polar o cilíndrico con el ángulo guardado en formato HMS.
- esférico esférico con los ángulos marcados para la visualización HMS (este es un formato solo de salida) esférico con los ángulos guardados en formato HMS.

La entrada puede ser en forma de 2 o 3 números, una lista con 2 o 3 números, un número complejo, un vector de 2 o 3 números o una matriz con n filas de 2 o 3 columnas.

La salida normalmente será en una forma similar a la entrada, con la excepción de números en bruto donde la salida será una lista.

0958 Función coseno

ID 958 · título recuperado de la página española

Sintaxis

`COS(Valor)`

Devuelve el coseno de Valor.

El valor se interpreta como radianes, grados o gradientes, según la configuración de la medición de ángulo en la Configuración de inicio, Configuración CAS o Configuración simbólica.

Ejemplos

[Ejemplo gráfico E1] (en modo grados)
 [Ejemplo gráfico E2]
 [Ejemplo gráfico E3] (en modo grados)
 [Ejemplo gráfico E4]

0563 Coseno hiperbólico

ID 563 · título recuperado de la página española

Sintaxis

`COSH(valor)`

Coseno hiperbólico

0558 Cotangente

ID 558 · título recuperado de la página española

Sintaxis

`COT(valor)`

Cotangente: $\text{COS}(X)/\text{SIN}(X)$

Ejemplos

[Ejemplo gráfico E1] (en modo grados)
 [Ejemplo gráfico E2]
 [Ejemplo gráfico E3] (en modo grados)
 [Ejemplo gráfico E4]

0657 Producto vectorial

ID 657 · título recuperado de la página española

Sintaxis

`CROSS(Vector1, Vector2)`

Devuelve el producto cruz de dos vectores.

0554 Cosecante

ID 554 · título recuperado de la página española

Sintaxis

CSC(valor)

Cosecante: 1/SIN(X)

Ejemplos

[Ejemplo gráfico E1] (en modo grados)
[Ejemplo gráfico E2]
[Ejemplo gráfico E3] (en modo grados)
[Ejemplo gráfico E4]

0966 C→PX — Convierte las coordenadas cartesianas en coordenadas de pantalla.

ID 966 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: C→PX

Explicación española: Convierte las coordenadas cartesianas en coordenadas de pantalla.

Sintaxis

C→PX(x, y)
C→PX({x, y})

Convierte las coordenadas cartesianas en coordenadas de pantalla.

Ejemplos

[Ejemplo gráfico E1] (suponiendo que la Configuración de gráfico actual de la aplicación está configurada como predeterminada)
[Ejemplo gráfico E2] (suponiendo que la Configuración de gráfico actual de la aplicación está configurada como predeterminada)

0809 Funciones en la vista de Inicio (PPL): 'D'

ID 809 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'D'

1018 Adición de fechas

ID 1018 · título recuperado de la página española

Sintaxis

DATEADD(Date, DayOffset)

Suma DayOffset días a Date y devuelve la fecha resultante en formato YYYY.MMDD.

Un DayOffset positivo adelanta la fecha; un DayOffset negativo la retrasa.

Date y DayOffset pueden ser escalares o listas. Si ambos son listas, DATEADD se aplica elemento a elemento. Si un argumento es escalar y el otro es una lista, el escalar se aplica a cada elemento de la lista.

1019 DAYOFWEEK — Día de la semana.

ID 1019 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: DAYOFWEEK

Explicación española: Día de la semana.

Sintaxis

DAYOFWEEK(Fecha)

Día de la semana. Dada una fecha en formato AAAA.MMDD, devuelve un número entre 1 (lunes) y 7 (domingo), que representa el día de la semana asociado con la fecha.

Ejemplos

[Ejemplo gráfico E1] (jueves)
[Ejemplo gráfico E2]

1017 Diferencia de fecha

ID 1017 · título recuperado de la página española

Sintaxis

DDAYS(Date1, Date2)

Calcula el número de días entre Date1 y Date2, donde ambas fechas se expresan en formato YYYY.MMDD.

El resultado es negativo si Date1 es posterior a Date2, y 0 si las fechas son iguales.

Date1 y Date2 pueden ser escalares o listas. Si ambas son listas, DDAYS se aplica elemento por elemento. Si un argumento es un escalar y el otro una lista, el escalar se aplica a cada elemento de la lista.

0967 Comando Debug (depuración)

ID 967 · título recuperado de la página española

Sintaxis

DEBUG(ProgramName(argumentos))
DEBUG

Inserta un punto de interrupción en un programa y activa el entorno de depuración. Cuando se encuentra la línea DEBUG; en un programa, el depurador se abre en la siguiente línea de código.

También puede utilizar este comando en la Vista de Inicio para depurar un programa. DEBUG(nombre) abre el depurador con el nombre del programa.

0070 Eliminar archivos de la aplicación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Eliminar archivos de la aplicación.](#)

0067 Eliminar aplicación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Eliminar aplicación.](#)

0074 Eliminar las variables de la aplicación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Eliminar las variables de la aplicación.](#)

0968 Eliminar columna

ID 968 · título recuperado de la página española

Sintaxis

DELCOL(matrix, column_number)

Elimina la columna column_number de la matriz matrix.

La matriz se modifica directamente. No es necesario asignar de nuevo el resultado a la variable.

1398 Eliminar las variables de inicio

ID 1398 · título recuperado de la página española

Sintaxis

DeLHVars(n)
DeLHVars("name")

Borra la variable de usuario de inicio especificada.

0970 Eliminar fila

ID 970 · título recuperado de la página española

Sintaxis

```
DELR0W(matrix, row_number)
```

Elimina la fila `row_number` de la matriz `matrix`.

La matriz se modifica directamente. No es necesario asignar de nuevo el resultado a la variable.

0617 Determinante de la matriz cuadrada

ID 617 · título recuperado de la página española

Sintaxis

```
DET(matriz)
```

Determinante de una matriz cuadrada.

0612 Diferencia de lista

ID 612 · título recuperado de la página española

Sintaxis

```
DIFFERENCE(List1 [, List2, ...])
```

Devuelve una lista de los elementos que no son comunes entre dos o más listas.

0973 Dimensiones de la cadena

ID 973 · título recuperado de la página española

Sintaxis

```
DIM(Object)  
DIM(List)
```

Si `Object` es una matriz, devuelve las dimensiones. Si `Object` es una cadena, devuelve el número de caracteres. Para una lista, devolverá una lista de tamaños de los elementos individuales.

0974 Dimensiones de GROB

ID 974 · título recuperado de la página española

Sintaxis

```
DIMGROB(G, w, h [, color])  
DIMGROB(G, w, h, list)
```

Establece el tamaño del GROB `G` en una anchura `w` y una altura `h`, en unidades cartesianas, e inicializa su contenido con un único color o con datos gráficos.

Parámetros

- `G`: GROB que se va a definir o redefinir.
- `w`: Anchura de `G` en unidades cartesianas.
- `h`: Altura de `G` en unidades cartesianas.
- `color`: Color de relleno opcional utilizado para inicializar `G`.
- `list`: Lista de datos gráficos utilizada para inicializar `G`.

Inicialización con un color

`DIMGROB(G, w, h [, color])` crea o redefine `G` como un GROB con una anchura `w` y una altura `h` en unidades cartesianas y lo rellena con color.

El componente alfa de color controla la transparencia de los píxeles inicializados y puede representar opacidad total, transparencia total o cualquier nivel admitido de transparencia parcial. Cuando el GROB se copia

posteriormente sobre otro GROB, por ejemplo mediante BLIT, los píxeles inicializados afectan al destino según su nivel de transparencia.

Inicialización con datos gráficos

DIMGROB(G, w, h, list) crea o redefine G como un GROB con una anchura w y una altura h en unidades cartesianas y lo inicializa mediante los datos gráficos almacenados en list.

El argumento list debe contener enteros binarios de 64 bits. Según el tamaño de los datos de imagen, cada entero codifica:

- 4 píxeles con una profundidad de color de 16 bits, o
- 2 píxeles con una profundidad de color de 32 bits.

Formatos de codificación de píxeles

En la lista de datos gráficos, los píxeles se codifican de la forma siguiente:

- Profundidad de color de 16 bits: Formato A1R5G5B5, que utiliza 1 bit para alfa y 5 bits para cada uno de los componentes rojo, verde y azul.
- Profundidad de color de 32 bits: Formato A8R8G8B8, que utiliza 8 bits para cada uno de los componentes alfa, rojo, verde y azul.

El formato de color de 32 bits se utiliza cuando el número total de píxeles, w*h, es exactamente el doble del número de enteros de list. En caso contrario, se presupone el formato de color de 16 bits.

El componente alfa controla la transparencia de cada píxel. Con una profundidad de color de 16 bits, su único bit solo permite dos estados: el valor 0 representa un píxel completamente opaco y el valor 1 representa un píxel completamente transparente. Con una profundidad de color de 32 bits, su valor de 8 bits admite transparencia parcial: 0x00 representa un píxel completamente opaco, 0xFF representa un píxel completamente transparente y los valores intermedios representan los niveles correspondientes de transparencia parcial.

Como se empaquetan varios píxeles en un único entero, los píxeles de cada entero se codifican en orden inverso con respecto a su aparición en la pantalla.

Ejemplo

Para crear una imagen de 1 fila y 4 píxeles con una profundidad de color de 32 bits, en el orden rojo, verde, azul y negro, se necesitan dos enteros de 64 bits.

El primer entero codifica los píxeles rojo y verde. Como los píxeles se almacenan en orden inverso dentro del entero, se escribe de la forma siguiente:

#0000FF0000FF0000:64h

El segundo entero codifica los píxeles azul y negro:

#00000000000000FF:64h

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

0975 Dimensiones de GROB

ID 975 · título recuperado de la página española

Sintaxis

DIMGROB_P(G, w, h [, color])
DIMGROB_P(G, w, h, list)

Establece el tamaño del GROB G en una anchura w y una altura h, en píxeles, e inicializa su contenido con un único color o con datos gráficos.

Parámetros

- G: GROB que se va a definir o redefinir.
- w: Anchura de G en píxeles.
- h: Altura de G en píxeles.
- color: Color de relleno opcional utilizado para inicializar G.
- list: Lista de datos gráficos utilizada para inicializar G.

Inicialización con un color

DIMGROB_P(G, w, h [, color]) crea o redefine G como un GROB de w por h píxeles y lo rellena con color.

El componente alfa de color controla la transparencia de los píxeles inicializados y puede representar opacidad total, transparencia total o cualquier nivel admitido de transparencia parcial. Cuando el GROB se copia posteriormente sobre otro GROB, por ejemplo mediante BLIT_P, los píxeles inicializados afectan al destino según su nivel de transparencia.

Inicialización con datos gráficos

DIMGROB_P(G, w, h, list) crea o redefine G como un GROB de w por h píxeles y lo inicializa mediante los datos gráficos almacenados en list.

El argumento list debe contener enteros binarios de 64 bits. Según el tamaño de los datos de imagen, cada entero codifica:

- 4 píxeles con una profundidad de color de 16 bits, o
- 2 píxeles con una profundidad de color de 32 bits.

Formatos de codificación de píxeles

En la lista de datos gráficos, los píxeles se codifican de la forma siguiente:

- Profundidad de color de 16 bits: Formato A1R5G5B5, que utiliza 1 bit para alfa y 5 bits para cada uno de los componentes rojo, verde y azul.
- Profundidad de color de 32 bits: Formato A8R8G8B8, que utiliza 8 bits para cada uno de los componentes alfa, rojo, verde y azul.

El formato de color de 32 bits se utiliza cuando el número total de píxeles, w*h, es exactamente el doble del número de enteros de list. En caso contrario, se presupone el formato de color de 16 bits.

El componente alfa controla la transparencia de cada píxel. Con una profundidad de color de 16 bits, su único bit solo permite dos estados: el valor 0 representa un píxel completamente opaco y el valor 1 representa un píxel completamente transparente. Con una profundidad de color de 32 bits, su valor de 8 bits admite transparencia parcial: 0x00 representa un píxel completamente opaco, 0xFF representa un píxel completamente transparente y los valores intermedios representan los niveles correspondientes de transparencia parcial.

Como se empaquetan varios píxeles en un único entero, los píxeles de cada entero se codifican en orden inverso con respecto a su aparición en la pantalla.

Ejemplo

Para crear una imagen de 1 fila y 4 píxeles con una profundidad de color de 32 bits, en el orden rojo, verde, azul y negro, se necesitan dos enteros de 64 bits.

El primer entero codifica los píxeles rojo y verde. Como los píxeles se almacenan en orden inverso dentro del entero, se escribe de la forma siguiente:

```
#0000FF0000FF0000:64h
```

El segundo entero codifica los píxeles azul y negro:

```
#00000000000000FF:64h
```

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

0658 Producto escalar

ID 658 · título recuperado de la página española

Sintaxis

```
DOT(Vector1, Vector2)
```

Devuelve el producto escalar de dos vectores.

0978 Dibujar menú de botón

ID 978 · título recuperado de la página española

Sintaxis

```
DRAWMENU(Object_1 [, ..., Object_6])
```

Dibuja un menú de seis botones en la parte inferior de la pantalla. Cada botón se puede etiquetar utilizando una cadena, una variable gráfica (G0-G9) o un nombre de ícono (ver la palabra clave ICON para más detalles). En el caso de números y otros tipos de objetos, se convertirán en una representación de cadena de acuerdo con la

configuración de inicio actual. Si el contenido del objeto proporcionado excede el tamaño del botón del menú, se truncará para ajustarse.

Ejemplos

[Ejemplo gráfico E1] crea un menú con el primer y tercer botón etiquetados como ABC y DEF, respectivamente. Las otras cuatro teclas del menú están en blanco.

0810 Funciones en la vista de Inicio (PPL): 'E'

ID 810 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'E'

0979 Base de logaritmo natural

ID 979 · título recuperado de la página española

Sintaxis

e

La constante matemática e (número de Euler), internamente representada como 2,71828182846

0980 Editar lista

ID 980 · título recuperado de la página española

Sintaxis

EDITLIST(list [, title [, read_only]])

Permite al usuario editar la lista especificada y devuelve la lista editada al finalizar.

Si se pasa una variable de lista como primer argumento, la variable se actualiza directamente cuando se hace clic en OK. No se requiere una asignación explícita.

El parámetro title puede ser:

- "Título"
- {"Título", {"Nombres de fila", ...}, {"Nombres de columna", ...}}.

La cadena "Título" se muestra encima del editor como título o nombre.

Si se especifican nombres de fila y de columna, se utilizan como encabezados de fila y columna.

Si read_only es distinto de cero, la lista se muestra en modo de solo lectura.

Ejemplos

[Ejemplo gráfico E1] edita la lista L1

[Ejemplo gráfico E2] muestra una lista pero no permite la edición

0981 Editar matriz

ID 981 · título recuperado de la página española

Sintaxis

EDITMAT(matrix [, title [, read_only]])

Permite al usuario editar la matriz especificada y devuelve la matriz editada al finalizar.

Si se pasa una variable de matriz como primer argumento, la variable se actualiza directamente cuando se hace clic en OK. No se requiere una asignación explícita.

El parámetro title puede ser:

- "Título"
- {"Título", {"Nombres de fila", ...}, {"Nombres de columna", ...}}.

La cadena "Título" se muestra encima del editor como título o nombre.

Si se especifican nombres de fila y de columna, se utilizan como encabezados de fila y columna.

Si read_only es distinto de cero, la matriz se muestra en modo de solo lectura.

Si se utiliza en un programa, la ejecución se reanuda cuando el usuario toca OK.

Ejemplos

[Ejemplo gráfico E1] edita la matriz M1.

1014 Editar matriz

ID 1014 · título recuperado de la página española

Sintaxis

EDITMAT(matrix [, title [, read_only]])

Permite al usuario editar la matriz especificada y devuelve la matriz editada al finalizar.

Si se pasa una variable de matriz como primer argumento, la variable se actualiza directamente cuando se hace clic en OK. No se requiere una asignación explícita.

El parámetro title puede ser:

- "Título"
- {"Título", {"Nombres de fila", ...}, {"Nombres de columna", ...}}.

La cadena "Título" se muestra encima del editor como título o nombre.

Si se especifican nombres de fila y de columna, se utilizan como encabezados de fila y columna.

Si read_only es distinto de cero, la matriz se muestra en modo de solo lectura.

Si se utiliza en un programa, la ejecución se reanuda cuando el usuario toca OK.

Ejemplos

[Ejemplo gráfico E1] edita la matriz M1.

0638 Valores propios

ID 638 · título recuperado de la página española

Sintaxis

EIGENVAL(matriz)

Muestra los valores propios en forma de vector para la matriz.

0639 Valores y vectores propios

ID 639 · título recuperado de la página española

Sintaxis

EIGENVV(matriz)

Vectores y valores propios de una matriz cuadrada

Muestra una lista de dos matrices. La primera contiene los vectores propios y la segunda los valores propios.

0987 END — Termina una estructura, un bloque, una prueba, un bucle o una bifurcación.

ID 987 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: END

Explicación española: Termina una estructura, un bloque, una prueba, un bucle o una bifurcación.

Termina una estructura, un bloque, una prueba, un bucle o una bifurcación.

1518 EQ — Devuelve 1 si los dos objetos son lo mismo.

ID 1518 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: EQ

Explicación española: Devuelve 1 si los dos objetos son lo mismo.

Sintaxis

EQ(objeto1, objeto2)

Devuelve 1 si los dos objetos son lo mismo.

Esta función es equivalente a la función `= y ==` con una excepción, si los dos objetos son listas, devuelve 1 si las dos listas son iguales, mientras que las funciones `= y ==` devuelven una lista que contiene 0 o 1 para cada par de elementos.

1231 error — Genera la visualización de un mensaje de error que contiene Cadena en un programa de CAS.

ID 1231 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: error

Explicación española: Genera la visualización de un mensaje de error que contiene Cadena en un programa de CAS.

Sintaxis

error(Cadena)

Genera la visualización de un mensaje de error que contiene Cadena en un programa de CAS.

0988 Evaluar la expresión

ID 988 · título recuperado de la página española

Sintaxis

EVAL(Expr)

Útil en los programas donde se pasan los parámetros no evaluados con el comando QUOTE.

1015 Evaluar la lista

ID 1015 · título recuperado de la página española

Sintaxis

EVALLIST({lista})

Evalúa el contenido de cada elemento en la lista y devuelve la lista resultante.

0994 Ejecutar sobre el elemento

ID 994 · título recuperado de la página española

Sintaxis

EXECON("&Expr", Lista1 [, Lista2, ...])

Crea una lista nueva evaluando una expresión en los elementos de una o más listas.

La expresión se escribe como una cadena de texto y usa referencias con ampersand (&) para indicar qué elementos se utilizan.

Con una lista, &1 indica el elemento actual. Las referencias &2, &3, ..., &9 indican los elementos siguientes según su posición relativa en la misma lista. La operación avanza por la lista y se detiene cuando ya no hay más grupos completos de elementos referenciados disponibles.

Con más de una lista, &1, &2, ... indican el elemento actual en la primera lista, en la segunda lista, y así sucesivamente. La operación avanza por las listas y se detiene cuando ya no hay más grupos completos de elementos referenciados disponibles.

Por ejemplo, en la expresión `"(&1+&2)/2"`, &1 se refiere al elemento actual en la primera lista y &2 se refiere al elemento actual en la segunda lista. La expresión calcula el promedio de cada par de elementos correspondientes.

Con más de una lista, también se pueden usar referencias de dos dígitos. En &ij, el primer dígito i indica el número de lista, en orden de izquierda a derecha, y el segundo dígito j indica la posición relativa del elemento en esa lista.

El segundo dígito puede ser de 1 a 9.

Por ejemplo, en la expresión "&23+&1", &23 se refiere al tercer elemento, relativo a la operación actual, en la segunda lista, y &1 se refiere al elemento actual en la primera lista. Los dos elementos referenciados se suman y, luego, la operación avanza al siguiente grupo de elementos disponible.

EXECON también puede operar con matrices de la misma manera que con listas.

0995 Exponencial natural

ID 995 · título recuperado de la página española

Sintaxis

EXP(valor)

Exponencial natural: e^x (antilogaritmo natural)

Devuelve el resultado de elevar e (número de Euler) a la potencia valor.

0551 Exponente menos 1

ID 551 · título recuperado de la página española

Sintaxis

EXPM1(valor)

Exponente menos 1: $(e^x)-1$

Esto es más preciso que EXP cuando x es cercana a cero.

1001 Función o variables EXPORT

ID 1001 · título recuperado de la página española

Sintaxis

```
EXPORT Var1[:=Val1] [, Var2[:=Val2], ...];  
EXPORT CONST Var1:=Val1 [, Var2:=Val2, ...];
```

EXPORT FunctionName[(Parameters)]

BEGIN

statements;

END;

EXPORT declara variables, constantes o funciones a nivel de programa como símbolos exportados.

Un símbolo exportado pasa a ser accesible globalmente y puede utilizarse desde fuera del programa mediante su nombre no cualificado.

Variables

Cuando se aplica a variables, EXPORT hace públicas las variables y constantes a nivel de programa.

Se puede acceder a una variable exportada directamente por su nombre desde fuera del programa y, cuando se pulsa la tecla Vars, esta aparece en el menú correspondiente de [[Inicio]], [[CAS]], [[Apl.]] o [[Usua.]], según el contexto.

Una variable o constante puede inicializarse al exportarse. Use := o = para asignar el valor inicial; ambas formas son equivalentes en este contexto.

Ejemplos de variables exportadas:

```
EXPORT ratio:=0.15, volume=9;
```

```
EXPORT CONST MaxSize=100;
```

```
EXPORT counter; // inicializada a 0 de forma predeterminada
```

Una única declaración de variables con EXPORT puede contener hasta 8 variables. Use instrucciones EXPORT adicionales para más variables.

Las variables a nivel de programa declaradas sin EXPORT son locales al programa de forma predeterminada.

Para obtener más información sobre el momento de la inicialización y la persistencia en memoria, toque el menú [[Otro]] y seleccione «Variables de PPL».

Funciones

Cuando se aplica a una definición de función, EXPORT hace pública la función y hace que aparezca en el Menú [[Usua.]] del cuadro de herramientas.

Una función exportada puede llamarse directamente desde fuera del programa mediante su nombre no cualificado, por ejemplo desde otro programa, una app o la Vista de Inicio:

FunctionName()

Ejemplo de una función exportada:

```
EXPORT X2M1(X)
```

```
BEGIN
```

```
RETURN X^2-1;
```

```
END;
```

También se admite la forma heredada de declaración anticipada `EXPORT FunctionName(Parameters);` por compatibilidad con programas más antiguos, pero el código nuevo normalmente debería definir directamente la función exportada.

Si no se especifica EXPORT para una función, la función es local al programa de forma predeterminada.

1002 Evaluar cadena

ID 1002 · título recuperado de la página española

Sintaxis

EXPR(Cadena)

Analiza una cadena, la convierte en un número o expresión y devuelve el resultado evaluado.

0811 Funciones en la vista de Inicio (PPL): 'F'

ID 811 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'F'

1101 Dibujar polígono relleno

ID 1101 · título recuperado de la página española

Sintaxis

FILLPOLY([G,] Coordinates [, Color [, alpha]])

Dibuja un polígono relleno en el GROB G mediante coordenadas cartesianas.

El polígono se define mediante una secuencia de vértices y se rellena utilizando Color con la transparencia opcional especificada mediante alpha.

Parámetros

- G: GROB de destino.
- Coordinates: Lista, vector o matriz que define los vértices del polígono.
- Color: Color opcional utilizado para rellenar el polígono.
- alpha: Transparencia opcional del color de relleno, de 0 (completamente opaco) a 255 (completamente transparente).

Cuando se especifica alpha, este sustituye el componente alfa de Color.

Definición de los vértices

En su forma más sencilla, Coordinates es un vector de valores numéricos consecutivos:

[x1, y1, x2, y2, x3, y3, ...]

Un vector debe contener un número par de elementos, ya que cada vértice se representa mediante un par de coordenadas x, y.

Cuando Coordinates es una matriz, sus elementos se interpretan como valores numéricos consecutivos que forman pares de coordenadas x, y. Una matriz puede tener cualquier número de filas y columnas, pero su número total de elementos debe ser par.

Cuando Coordinates es una lista, se aplican las reglas generales para especificar coordenadas y puntos.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1100 Dibujar polígono relleno

ID 1100 · título recuperado de la página española

Sintaxis

```
FILLPOLY_P([G,] Coordinates [, Color [, alpha]])
```

Dibuja un polígono relleno en el GROB G mediante coordenadas de píxel.

El polígono se define mediante una secuencia de vértices y se rellena utilizando Color con la transparencia opcional especificada mediante alpha.

Parámetros

- G: GROB de destino.
- Coordinates: Lista, vector o matriz que define los vértices del polígono.
- Color: Color opcional utilizado para rellenar el polígono.
- alpha: Transparencia opcional del color de relleno, de 0 (completamente opaco) a 255 (completamente transparente).

Cuando se especifica alpha, este sustituye el componente alfa de Color.

Definición de los vértices

En su forma más sencilla, Coordinates es un vector de valores numéricos consecutivos:

```
[x1, y1, x2, y2, x3, y3, ...]
```

Un vector debe contener un número par de elementos, ya que cada vértice se representa mediante un par de coordenadas x, y.

Cuando Coordinates es una matriz, sus elementos se interpretan como valores numéricos consecutivos que forman pares de coordenadas x, y. Una matriz puede tener cualquier número de filas y columnas, pero su número total de elementos debe ser par.

Cuando Coordinates es una lista, se aplican las reglas generales para especificar coordenadas y puntos.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

0581 Densidad de Fisher

ID 581 · título recuperado de la página española

Sintaxis

```
FISHER(n, d, x)
```

Función de densidad de probabilidad F (Fisher o Fisher-Snedecor).

Calcula la densidad de probabilidad en el valor x, con grados de libertad en el numerador n y el denominador d.

0589 Fisher acumulada

ID 589 · título recuperado de la página española

Sintaxis

```
FISHER_CDF(n, d, x [, x2])
```

Función de distribución F acumulada (Fisher o Fisher-Snedecor)

Devuelve la probabilidad de cola inferior de la función de densidad de probabilidad de Fisher para el valor x, con grados de libertad de numerador n y de denominador d. Con el cuarto argumento opcional x2, devuelve el área bajo la función de densidad de probabilidad de Fisher entre los dos valores de x.

0597 Fisher acumulada inversa

ID 597 · título recuperado de la página española

Sintaxis`FISHER_ICDF(n, d, p)`

Función de distribución acumulativa inversa F (Fisher o Fisher-Snedecor).

Devuelve el valor x de modo tal que la probabilidad de la cola inferior F de x, con grados de libertad en el numerador n, y el denominador d, sea p.

0530 Suelo

ID 530 · título recuperado de la página española

Sintaxis`FL00R(valor)`

Entero más grande menor o igual que valor.

0541 Encontrar raíz

ID 541 · título recuperado de la página española

Sintaxis`FNR00T(Expr, Var [, supuesto [, supuesto2]])`

Encontrar raíz (como la aplicación Soluc).

Busca el valor de la variable indicada en que la expresión se aproxima más a cero. Utiliza supuesto como estimación inicial.

1027 Estructura de bucle For

ID 1027 · título recuperado de la página española

Sintaxis`FOR var FROM inicio TO (o DOWNTO) fin [STEP incremento] DO comandos END;`

Establece la variable var en inicio; luego, mientras el valor de esta variable sea igual o menor que (o más que en un DOWNTO) fin, ejecuta comandos y suma (o resta DOWNTO) 1 (o incremento) a var.

Código de ejemplo:

```
//print 1 3 5 7 9
FOR A FROM 1 TO 10 STEP 2 DO
PRINT(A);
END;

//print 10 8 6 4 2
FOR A FROM 10 DOWNTO 1 STEP 2 DO
PRINT(A);
END;
```

0532 Parte fraccional

ID 532 · título recuperado de la página española

Sintaxis`FP(valor)`

Devuelve la parte fraccionaria del valor.

1031 Congelación de pantalla

ID 1031 · título recuperado de la página española

Sintaxis`FREEZE`

Evita que la pantalla se vuelva a dibujar tras la finalización del programa. Deja la presentación modificada en pantalla para que la vea el usuario.

Este comando no se detiene y espera entradas. En vez de ello, evita tener que redibujar hasta que otra operación (presionar una tecla, tocar la pantalla o un comando o comunicación de datos) activa la pantalla que se va a dibujar.

0812 Funciones en la vista de Inicio (PPL): 'G'

ID 812 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'G'

1733 GET — Recupera un objeto de un objeto composite (lista, matriz, cadena o entero binario) utilizando el índice...

ID 1733 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: GET

Explicación española: Recupera un objeto de un objeto composite (lista, matriz, cadena o entero binario) utilizando el índice...

Sintaxis

GET(composite, index1 [, ..., indexN])

Recupera un objeto de un objeto composite (lista, matriz, cadena o entero binario) utilizando el índice o los índices especificados por index1 hasta indexN.

Esto funciona de la misma manera que el acceso a variables, pero permite trabajar con objetos que no están almacenados en una variable.

1039 Obtener Base

ID 1039 · título recuperado de la página española

Sintaxis

GETBASE(#entero[m])

Devuelve el número base de entero con el marcador de base m. La función SETBASE utiliza el número base.

0 = Sistema

1 = Binario

2 = Octal

3 = Decimal

4 = Hexadecimal

El marcador de base m puede ser b (para binario), o (para octal), d (para decimal) o h (para hexadecimal). Si se omite m, se supone la base del sistema actual.

Ejemplos

[Ejemplo gráfico E1]

[Ejemplo gráfico E2] (si la base predeterminada es hexadecimal)

[Ejemplo gráfico E3]

1040 Obtener bits

ID 1040 · título recuperado de la página española

Sintaxis

GETBITS(#entero)

Devuelve la cantidad de bits usados para codificar un entero. Si no se especifica, se usa el valor del campo Enteros de la Página 1 de las configuraciones de Inicio.

Ejemplos

[Ejemplo gráfico E1] (Si los enteros de la Configuración de inicio se establecen en 32 bits)
[Ejemplo gráfico E2]
[Ejemplo gráfico E3]
[Ejemplo gráfico E4]

1041 Obtener tecla

ID 1041 · título recuperado de la página española

Sintaxis

GETKEY

Devuelve el ID de la primera tecla del búfer de teclado o -1 si no se ha pulsado ninguna tecla desde la última llamada a GETKEY. Los ID de tecla son enteros de 0 a 50, numerados desde la esquina superior izquierda (tecla 0) a la esquina inferior derecha (tecla 50).

0 = Apps
1 = Symb
2 = Arriba
3 = Help
4 = Esc
5 = Inicio
6 = Plot
7 = Izquierda
8 = Derecha
9 = View
10 = CAS
11 = Num
12 = Abajo
13 = Menu

Después de eso, las teclas se numeran de arriba a la izquierda (14 = Vars) a la parte inferior derecha (50 = +)

1042 Obtener píxel de color

ID 1042 · título recuperado de la página española

Sintaxis

GETPIX([G,] x, y)

Devuelve el color del píxel situado en la posición especificada del GROB G mediante coordenadas cartesianas.

Parámetros

- G: GROB de origen.
- x, y: Coordenadas cartesianas del píxel cuyo color se devuelve.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1043 Obtener píxel de color

ID 1043 · título recuperado de la página española

Sintaxis

GETPIX_P([G,] x, y)

Devuelve el color del píxel situado en la posición especificada del GROB G mediante coordenadas de píxel.

Parámetros

- G: GROB de origen.
- x, y: coordenadas de píxel del píxel cuyo color se devuelve.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1044 Crear campo de Galois

ID 1044 · título recuperado de la página española

Sintaxis

GF(Enterop, Enteron)

Crea un campo de Galois de p característico con p^n elementos.

1046 Altura GROB

ID 1046 · título recuperado de la página española

Sintaxis

GROBH([G])

Devuelve la altura del GROB G en unidades cartesianas, medida a lo largo del eje y y de abajo hacia arriba. Si se omite G , se utiliza G_0 .

1047 Altura GROB

ID 1047 · título recuperado de la página española

Sintaxis

GROBH_P([G])

Devuelve la altura del GROB G en píxeles. Si se omite G , se utiliza G_0 .

1048 Ancho de GROB

ID 1048 · título recuperado de la página española

Sintaxis

GROBW([G])

Devuelve el ancho del GROB G en unidades cartesianas, medido a lo largo del eje x de izquierda a derecha. Si se omite G , se utiliza G_0 .

1049 Ancho de GROB

ID 1049 · título recuperado de la página española

Sintaxis

GROBW_P([G])

Devuelve el ancho del GROB G en píxeles. Si se omite G , se utiliza G_0 .

0813 Funciones en la vista de Inicio (PPL): 'H'

ID 813 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'H'

1060 HMS→ — Muestra un valor sexagesimal en formato decimal.

ID 1060 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: HMS→

Explicación española: Muestra un valor sexagesimal en formato decimal.

Sintaxis

HMS→(valor)

Muestra un valor sexagesimal en formato decimal.

1059 →HMS — Muestra un valor decimal en formato sexagesimal.

ID 1059 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: →HMS

Explicación española: Muestra un valor decimal en formato sexagesimal.

Sintaxis

→HMS(valor)

Muestra un valor decimal en formato sexagesimal. Es decir, en unidades subdivididas en grupos de 60. Esto incluye grados, minutos y segundos, así como horas, minutos y segundos.

0814 Funciones en la vista de Inicio (PPL): 'I'

ID 814 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'I'

0621 Matriz de identidad

ID 621 · título recuperado de la página española

Sintaxis

IDENMAT(n)

Crea una matriz cuadrada de la dimensión n x n cuyos elementos diagonales son 1 y los elementos fuera de la diagonal son cero.

1069 Estructura de bifurcación IF

ID 1069 · título recuperado de la página española

Sintaxis

IF prueba THEN comandos1 [ELSE comandos2] END;

Empieza con la estructura de bifurcación "SI ... ENTONCES ... FIN" o "SI ... ENTONCES... SI NO... FIN".

Evalúa la prueba. Si la prueba es verdadera (no 0), ejecuta los comandos1, de lo contrario, ejecuta los comandos2

Código de ejemplo:

```
IF A<1 THEN
PRINT("A<1");
ELSE
PRINT("A>1");
END;
```

1070 Estructura de la captura de error

ID 1070 · título recuperado de la página española

Sintaxis

IFERR commands1 THEN commands2 [ELSE commands3] END;

IFERR ejecuta la secuencia protegida de comandos especificada por commands1. Si ocurre un error al ejecutar commands1, IFERR ejecuta la secuencia de manejo de errores especificada por commands2. Si no ocurre ningún error y la cláusula ELSE opcional está presente, IFERR ejecuta la secuencia especificada por commands3.

Muchas condiciones son reconocidas automáticamente por la HP Prime como condiciones de error y son tratadas como errores en los programas. IFERR facilita la captura y el manejo de dichos errores.

Nota: cuando un error es capturado por IFERR, el número de error se almacena en la variable Ans. Puede acceder a este valor en la cláusula THEN de IFERR.

Código de ejemplo:

```
IFERR 1/0 THEN
PRINT("1/0 Error");
END;
```

1071 Estructura Si Entonces Si no

ID 1071 · título recuperado de la página española

Sintaxis

```
IFTE(Expr, CláusulaVerdadera, CláusulaFalsa)
```

Si Expr evalúa verdadero (1), evalúa CláusulaVerdadera; si no, evalúa CláusulaFalsa.

Si Expr devuelve una lista, entonces CláusulaVerdadera y CláusulaFalsa tienen que ser un objeto único o una lista del mismo tamaño que el resultado de Expr. El resultado será una lista de ese tamaño con elementos seleccionados de CláusulaVerdadera y CláusulaFalsa según el valor booleano de cada elemento del resultado de Expr.

0546 Parte imaginaria

ID 546 · título recuperado de la página española

Sintaxis

```
IM(x+yi)
```

Devuelve la parte imaginaria de un número complejo.

1075 Formulario de entrada

ID 1075 · título recuperado de la página española

Sintaxis

```
INPUT(var [, "título" [, "etiqueta" [, "ayuda" [, restablecer_valor [, valor_inicial]]]])
INPUT({vars} [, "títulos" [, {"etiquetas"} [, {"ayudas"} [, {restablecer_valores} [, {valores_iniciales}]]]])
var -> {nombre var, real, [{pos}]}
var -> {nombre var, [tipos permitidos de matriz], [{pos}]}
var -> {nombre var, {elegir elementos}, [{pos}]}
```

La formula más simple de este comando abre un cuadro de diálogo con el título dado y un campo llamado etiqueta, además de que muestra la ayuda en la parte inferior. El cuadro de diálogo incluye las teclas de menú [[Canc.]] y [[OK]]. el usuario puede introducir un valor en el campo etiquetado. Si el usuario presiona la tecla de menú [[OK]], la variable var se actualiza con el valor introducido y se devuelve 1. Si el usuario presiona la tecla Esc o la tecla de menú [[Canc.]], no se actualiza var y se devuelve 0.

El argumento restablecer_valor es el valor que aparece cuando se presiona la tecla Backspace mientras el campo tiene el foco, pero no está en modo de edición.

Si se especifica valor_inicial, aparecerá como el valor inicial en el campo dado. De lo contrario, aparecerá el valor actual de la variable asignada.

En la forma más compleja del comando, las listas se utilizan para crear un cuadro de diálogo pluridisciplinar. Si var es una lista, cada elemento puede ser un nombre de variable o una lista utilizando el siguiente formato:

{nombre var, real, [{pos}]} para crear un control con casilla de verificación. Si real es >1, esta casilla de verificación se agrupa con la siguiente casilla de verificación de n -1 en un grupo de radio (por ej., solo una de las casillas de verificación n se pueden marcar en cualquier momento)

{nombre var, [tipos permitidos de matriz] , [{pos}]} para crear un campo de edición. tipos permitidos de matriz enumera todos los tipos permitidos ([-1] representa todos los tipos permitidos). Si el único tipo permitido es una cadena, entonces la edición ocultará las comillas dobles.

{nombre var, {elegir elementos}, [{pos}]} para crear un campo de selección.

Si se especifica pos, esta es una lista del formulario {el inicio del campo en el porcentaje de pantalla, el ancho del campo en el porcentaje de pantalla, la línea (empieza en 0)}. Esto permite que controle con precisión la posición y el tamaño de sus campos. Tome en cuenta que tiene que especificar pos para ninguno o todos los campos en el cuadro de diálogo.

Hay un máximo de 7 líneas de controles por página. Los controles con más de 7 líneas se colocarán en las siguientes páginas. Si se crea más de una página, los títulos pueden ser una lista de títulos.

1649 INSERT

ID 1649 · título localizado en HelpEs

Sintaxis

INSERT(objeto1, Índice, objeto2)

Inserta objeto2 en objeto1 inmediatamente antes de la posición especificada por Índice. Si Índice es mayor que el tamaño de objeto1, objeto2 se adjuntará a objeto1.

objeto1 puede ser una lista, un vector o una cadena. objeto2 puede ser cualquier cosa si objeto1 es una lista. objeto2 debe ser un número real o complejo si objeto1 es un vector. objeto2 debe ser una sola cadena de caracteres si objeto1 es una cadena.

1077 En cadena

ID 1077 · título recuperado de la página española

Sintaxis

INSTRING(String, Substring [, StartPos [, Occurrence]])

Devuelve la posición de la ocurrencia solicitada de la subcadena Substring en la cadena String o 0 si no se encuentra.

La posición inicial StartPos especifica la posición donde comienza la búsqueda. Si se omite, StartPos es 1.

El número de ocurrencia Occurrence especifica qué ocurrencia coincidente devolver, empezando desde StartPos. Si se omite, Occurrence es 1. Si la ocurrencia solicitada no se encuentra, INSTRING devuelve 0.

Cada argumento también puede ser una lista. En ese caso, INSTRING procesa la entrada elemento por elemento y devuelve una lista de resultados.

Cuando varios argumentos son listas, sus elementos se procesan por pares. Todos los argumentos de lista deben tener la misma longitud; de lo contrario, se produce un error. Cualquier argumento escalar se utiliza para cada elemento procesado.

0613 Lista de Intersección

ID 613 · título recuperado de la página española

Sintaxis

INTERSECT(List1, List2 [, List3, ...])

Devuelve una lista de elementos comunes en dos o más listas.

1080 GROB Invertida

ID 1080 · título recuperado de la página española

Sintaxis

INVERT([G] [, x1, y1 [, x2, y2]])

Invierte los colores de una región rectangular del GROB G mediante coordenadas cartesianas, produciendo un efecto de vídeo inverso.

Parámetros

- G: GROB que contiene la región que se va a invertir.
- x1, y1: Coordenadas cartesianas opcionales de la esquina superior izquierda de la región.
- x2, y2: Coordenadas cartesianas opcionales de la esquina inferior derecha de la región.

Si no se especifican coordenadas, se invierte todo el GROB.

Si solo se especifican x1 e y1, la región se extiende desde ese punto hasta la esquina inferior derecha de G.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1081 GROB Invertida

Sintaxis

```
INVERT_P([G] [, x1, y1 [, x2, y2]])
```

Invierte los colores de una región rectangular del GROB G mediante coordenadas de píxel, produciendo un efecto de vídeo inverso.

Parámetros

- G: GROB que contiene la región que se va a invertir.
- x1, y1: coordenadas de píxel opcionales de la esquina superior izquierda de la región.
- x2, y2: coordenadas de píxel opcionales de la esquina inferior derecha de la región.

Si no se especifican coordenadas, se invierte todo el GROB.

Si solo se especifican x1 e y1, la región se extiende desde ese punto hasta la esquina inferior derecha de G.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

0531 Parte entera**Sintaxis**

```
IP(valor)
```

Devuelve la parte entera de valor.

1089 Se presiona la tecla**Sintaxis**

```
ISKEYDOWN(IDTecla)
ISKEYDOWN(-1)
ISKEYDOWN()
```

Devuelve 1 (verdadero) si la tecla cuyo identificador IDTecla se proporciona está actualmente pulsada, y 0 (falso) en caso contrario.

Las teclas están numeradas desde la parte superior izquierda ([Apps] = 0) hasta la parte inferior derecha ([+] = 50).

Cuando ISKEYDOWN se llama con el argumento -1, devuelve un número entero de 64 bits cuyos 51 bits menos significativos corresponden al estado actual de las teclas individuales. Si la n-ésima tecla está pulsada, el n-ésimo bit de este número entero se establece. De este modo es posible comprobar simultáneamente el estado de varias teclas.

Cuando se llama sin argumentos, devuelve 1 (verdadero) si alguna tecla está pulsada en ese momento, o 0 (falso) si no hay ninguna tecla pulsada.

Nota: La tecla [On] es una tecla especial que interrumpe la ejecución del programa, y su estado no puede comprobarse con ISKEYDOWN.

1090 Iterar expresión**Sintaxis**

```
ITERATE(expr, var, valori, veces)
```

Evalúa expr en términos de var repetidamente para veces, empezando por var = valori.

0815 Funciones en la vista de Inicio (PPL): 'K'

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'K'

1508 Palabra clave de tecla

ID 1508 · título recuperado de la página española

Sintaxis

```
KEY nombre  
BEGIN  
END;
```

Especificar una función con la palabra clave de la KEY permite redefinir la tecla correspondiente en el teclado.

El nombre de la función especifica la tecla.

Consulte el manual del usuario para ver la lista completa.

Tenga en cuenta que una función de tecla también pueden exportarse y ser una vista.

1102 Detener ejecución

ID 1102 · título recuperado de la página española

Sintaxis

```
KILL;
```

Detiene la ejecución de un programa.

0816 Funciones en la vista de Inicio (PPL): 'L'

ID 816 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'L'

1108 Parte izquierda

ID 1108 · título recuperado de la página española

Sintaxis

```
LEFT(Cadena, Entero)
```

Dada una cadena y un entero n, devuelve los primeros caracteres n de la cadena. Si $n \geq \text{DIM}(\text{str})$ o $n \leq 0$, devuelve la cadena completa.

1113 Dibujo de líneas

ID 1113 · título recuperado de la página española

Sintaxis

```
LINE([G,] x1, y1, x2, y2 [, color])  
LINE([G,] points_definition, line_definitions, rotation_matrix or {rotation_matrix or -1, ["N"], [{eye_x, eye_y, eye_z}  
or -1], [{xmin3D, xmax3D, ymin3D, ymax3D, zmin3D, zmax3D}]} [, zstring])  
LINE([G,] pre_rotated_points, line_definitions [, zstring])
```

Dibuja una o varias líneas en el GROB G mediante coordenadas cartesianas.

La forma básica dibuja una sola línea entre dos puntos especificados. Las formas avanzadas permiten renderizar varias líneas en una sola operación y aplicar opcionalmente transformaciones geométricas, proyección en perspectiva, recorte 3D, normalización de profundidad y prueba de profundidad por píxel.

Parámetros

- G: GROB de destino.
- x1, y1: Coordenadas cartesianas del primer extremo.
- x2, y2: Coordenadas cartesianas del segundo extremo.
- color: Color opcional de la línea. En una definición de línea, este color sustituye los colores almacenados en las definiciones de punto correspondientes.
- alpha: Transparencia opcional aplicada a la línea, de 0 (completamente opaca) a 255 (completamente transparente).

- **zstring:** Cadena opcional de búfer de profundidad utilizada para la prueba de profundidad por píxel. Debe crearse mediante `TRIANGLE([G])` y se actualiza a medida que se dibujan las líneas.

Dibujo de una sola línea

`LINE([G,] x1, y1, x2, y2 [, color])` dibuja una línea desde (x1, y1) hasta (x2, y2) mediante coordenadas cartesianas y, si se especifica, utiliza el color indicado.

Renderizado de varias líneas

- **points_definition:** Lista o matriz que contiene los puntos utilizados por las líneas. Cada definición de punto contiene las coordenadas x e y y puede incluir además una coordenada z, un color o ambos. Las coordenadas también pueden proporcionarse como un único valor de punto.
- **line_definitions:** Lista o matriz que describe las líneas que se van a renderizar. Cada definición de línea contiene entre dos y cuatro valores con la forma `p1, p2 [, color [, alpha]]`.
- **p1, p2:** Índices de las dos entradas de `points_definition` que definen la línea.
- **color:** Color opcional que sustituye los colores almacenados en las definiciones de punto correspondientes. Para especificar alpha sin sustituir los colores de los puntos, utilice -1 para color.

Varias definiciones de línea pueden compartir el mismo color y transparencia mediante la forma agrupada `{color [, alpha], line_1, ..., line_n}`.

Transformación y proyección

- **rotation_matrix:** Matriz de transformación con dimensiones comprendidas entre 2x2 y 3x4. Cada punto se multiplica por esta matriz para aplicar la transformación geométrica correspondiente, incluida la traslación cuando la forma de la matriz lo permite.

El argumento de transformación también puede especificarse como una lista con la forma siguiente:

`{rotation_matrix or -1 [, "N"] [, {eye_x, eye_y, eye_z} or -1] [, {xmin3D, xmax3D, ymin3D, ymax3D, zmin3D, zmax3D}]}`

- **rotation_matrix or -1:** Matriz utilizada para transformar los puntos. Especifique -1 cuando no deba aplicarse ninguna matriz de transformación.
- **"N":** Indicador opcional de normalización. Cuando se especifica, las coordenadas z transformadas se normalizan al intervalo de 0 a 255, lo que permite utilizarlas para la prueba de profundidad.
- **{eye_x, eye_y, eye_z} or -1:** Posición opcional del observador utilizada para la proyección en perspectiva. Especifique -1 cuando no sea necesaria la proyección en perspectiva.
- **{xmin3D, xmax3D, ymin3D, ymax3D, zmin3D, zmax3D}:** Límites opcionales que definen la región de recorte 3D.

Después de la transformación, la proyección en perspectiva se aplica mediante las ecuaciones siguientes:

$$x = \text{eye_z}/z * x - \text{eye_x}$$

$$y = \text{eye_z}/z * y - \text{eye_y}$$

Si se proporcionan límites de recorte 3D, cada línea se recorta con respecto a la región especificada antes de renderizarse.

Reutilización de puntos transformados

Cuando se aplica una transformación, `LINE` devuelve una cadena que contiene los puntos transformados. Esta cadena puede pasarse posteriormente como `pre_rotated_points`, lo que permite volver a renderizar la misma geometría transformada sin repetir la transformación.

- **pre_rotated_points:** Cadena de puntos transformados devuelta por una llamada anterior a `LINE` o `TRIANGLE`.
- **line_definitions:** Definiciones de línea que hacen referencia a los puntos almacenados en `pre_rotated_points`.

La cadena de puntos transformados devuelta también puede reutilizarse mediante `TRIANGLE` cuando se necesiten los mismos puntos y la misma transformación en varias operaciones de dibujo consecutivas.

Prueba de profundidad

`TRIANGLE([G])` crea y devuelve una cadena de búfer de profundidad adaptada a las dimensiones de G. A cada píxel del búfer se le asigna inicialmente un valor de profundidad de 255.

Pase esta cadena como `zstring` junto con valores de profundidad de los puntos comprendidos entre 0 y 255. Durante el renderizado, no se dibujan los píxeles que estén más alejados que los píxeles ya registrados en el búfer. Los píxeles que superan la prueba de profundidad se dibujan y los valores correspondientes de `zstring` se actualizan automáticamente.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1114 Dibujo de líneas

ID 1114 · título recuperado de la página española

Sintaxis

```
LINE_P([G,] x1, y1, x2, y2 [, color])  
LINE_P([G,] points_definition, line_definitions, rotation_matrix or {rotation_matrix or -1, ["N"], [{eye_x, eye_y, eye_z} or -1], [{xmin3D, xmax3D, ymin3D, ymax3D, zmin3D, zmax3D}]} [, zstring])  
LINE_P([G,] pre_rotated_points, line_definitions [, zstring])
```

Dibuja una o varias líneas en el GROB G mediante coordenadas de píxel.

La forma básica dibuja una sola línea entre dos puntos especificados. Las formas avanzadas permiten renderizar varias líneas en una sola operación y aplicar opcionalmente transformaciones geométricas, proyección en perspectiva, recorte 3D, normalización de profundidad y prueba de profundidad por píxel.

Parámetros

- G: GROB de destino.
- x1, y1: coordenadas de píxel del primer extremo.
- x2, y2: coordenadas de píxel del segundo extremo.
- color: Color opcional de la línea. En una definición de línea, este color sustituye los colores almacenados en las definiciones de punto correspondientes.
- alpha: Transparencia opcional aplicada a la línea, de 0 (completamente opaca) a 255 (completamente transparente).
- zstring: Cadena opcional de búfer de profundidad utilizada para la prueba de profundidad por píxel. Debe crearse mediante TRIANGLE_P([G]) y se actualiza a medida que se dibujan las líneas.

Dibujo de una sola línea

LINE_P([G,] x1, y1, x2, y2 [, color]) dibuja una línea desde (x1, y1) hasta (x2, y2) mediante coordenadas de píxel y, si se especifica, utiliza el color indicado.

Renderizado de varias líneas

- points_definition: Lista o matriz que contiene los puntos utilizados por las líneas. Cada definición de punto contiene las coordenadas x e y y puede incluir además una coordenada z, un color o ambos. Las coordenadas también pueden proporcionarse como un único valor de punto.
- line_definitions: Lista o matriz que describe las líneas que se van a renderizar. Cada definición de línea contiene entre dos y cuatro valores con la forma p1, p2 [, color [, alpha]].
- p1, p2: Índices de las dos entradas de points_definition que definen la línea.
- color: Color opcional que sustituye los colores almacenados en las definiciones de punto correspondientes. Para especificar alpha sin sustituir los colores de los puntos, utilice -1 para color.

Varias definiciones de línea pueden compartir el mismo color y transparencia mediante la forma agrupada {color [, alpha], line_1, ..., line_n}.

Transformación y proyección

- rotation_matrix: Matriz de transformación con dimensiones comprendidas entre 2x2 y 3x4. Cada punto se multiplica por esta matriz para aplicar la transformación geométrica correspondiente, incluida la traslación cuando la forma de la matriz lo permite.

El argumento de transformación también puede especificarse como una lista con la forma siguiente:

```
{rotation_matrix or -1 [, "N"] [, {eye_x, eye_y, eye_z} or -1] [, {xmin3D, xmax3D, ymin3D, ymax3D, zmin3D, zmax3D}]}
```

- rotation_matrix or -1: Matriz utilizada para transformar los puntos. Especifique -1 cuando no deba aplicarse ninguna matriz de transformación.
- "N": Indicador opcional de normalización. Cuando se especifica, las coordenadas z transformadas se normalizan al intervalo de 0 a 255, lo que permite utilizarlas para la prueba de profundidad.
- {eye_x, eye_y, eye_z} or -1: Posición opcional del observador utilizada para la proyección en perspectiva. Especifique -1 cuando no sea necesaria la proyección en perspectiva.
- {xmin3D, xmax3D, ymin3D, ymax3D, zmin3D, zmax3D}: Límites opcionales que definen la región de recorte 3D.

Después de la transformación, la proyección en perspectiva se aplica mediante las ecuaciones siguientes:

```
x = eye_z/z*x-eye_x  
y = eye_z/z*y-eye_y
```

Si se proporcionan límites de recorte 3D, cada línea se recorta con respecto a la región especificada antes de renderizarse.

Reutilización de puntos transformados

Cuando se aplica una transformación, `LINE_P` devuelve una cadena que contiene los puntos transformados. Esta cadena puede pasarse posteriormente como `pre_rotated_points`, lo que permite volver a renderizar la misma geometría transformada sin repetir la transformación.

- `pre_rotated_points`: Cadena de puntos transformados devuelta por una llamada anterior a `LINE_P` o `TRIANGLE_P`.
- `line_definitions`: Definiciones de línea que hacen referencia a los puntos almacenados en `pre_rotated_points`.

La cadena de puntos transformados devuelta también puede reutilizarse mediante `TRIANGLE_P` cuando se necesiten los mismos puntos y la misma transformación en varias operaciones de dibujo consecutivas.

Prueba de profundidad

`TRIANGLE_P([G])` crea y devuelve una cadena de búfer de profundidad adaptada a las dimensiones de `G`. A cada píxel del búfer se le asigna inicialmente un valor de profundidad de 255.

Pase esta cadena como `zstring` junto con valores de profundidad de los puntos comprendidos entre 0 y 255. Durante el renderizado, no se dibujan los píxeles que estén más alejados que los píxeles ya registrados en el búfer. Los píxeles que superan la prueba de profundidad se dibujan y los valores correspondientes de `zstring` se actualizan automáticamente.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1735 ListToMat — Convierte una lista en un vector o una lista de listas en una matriz.

ID 1735 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `ListToMat`

Explicación española: Convierte una lista en un vector o una lista de listas en una matriz.

Sintaxis

```
ListToMat({})  
ListToMat({{ }, { }...})
```

Convierte una lista en un vector o una lista de listas en una matriz.

1120 Logaritmo natural

ID 1120 · título recuperado de la página española

Sintaxis

```
LN(valor)
```

Función logarítmica natural

Devuelve el logaritmo de valor en base e, número de Euler.

0552 Logaritmo natural más 1

ID 552 · título recuperado de la página española

Sintaxis

```
LNPI(valor)
```

Logaritmo natural más 1: $\text{LN}(X+1)$

Es más precisa que la función logaritmo natural cuando el valor es próximo a cero.

1123 Palabra clave LOCAL

ID 1123 · título recuperado de la página española

Sintaxis

```
[LOCAL] Var1[:=Val1] [, Var2[:=Val2], ...];  
[LOCAL] CONST Var1:=Val1 [, Var2:=Val2, ...];  
[LOCAL] FunctionName(Param1 [, Param2, ...])
```

LOCAL declara variables locales, constantes locales del programa o funciones locales del programa.

Variables

A nivel de programa, una variable o constante declarada sin EXPORT es local del programa, y la palabra clave LOCAL es opcional en este contexto.

Dentro de una función, LOCAL es necesario para declarar variables cuyo ámbito está limitado al bloque envolvente. Si se declara una variable con el mismo nombre en un bloque interno, se trata de una variable distinta y oculta la variable exterior solo dentro de ese bloque.

CONST solo puede usarse para constantes a nivel de programa. No puede usarse para variables declaradas dentro de una función.

Los valores iniciales pueden asignarse en la declaración usando := o =; ambas formas son equivalentes en este contexto.

Ejemplos de variables y constantes locales:

```
LOCAL width:=10, height=5, flags={};
```

```
LOCAL area; // inicializada en 0 de forma predeterminada
```

```
LOCAL CONST MaxSize=100;
```

Una única declaración de variables LOCAL puede contener hasta 8 variables. Use instrucciones LOCAL adicionales para más variables.

Las variables y constantes locales del programa pueden seguir siendo accesibles desde fuera del programa usando un prefijo cualificado con el nombre del programa, por ejemplo:

```
ProgramName.VariableName
```

Para obtener más información sobre el momento de la inicialización y la persistencia en memoria, toque el menú [[Otro]] y seleccione «Variables de PPL».

Funciones

A nivel de programa, LOCAL puede utilizarse para declarar funciones locales del programa.

Si no se especifica ni LOCAL ni EXPORT, la función es local del programa de forma predeterminada.

Una función local del programa puede llamarse directamente desde dentro del mismo programa. Aunque no puede llamarse directamente desde fuera del programa mediante su nombre no cualificado, puede seguir siendo accesible usando un prefijo cualificado con el nombre del programa, por ejemplo:

```
ProgramName.FunctionName()
```

Por tanto, LOCAL no proporciona una encapsulación privada completa. Solo impide el acceso directo mediante un nombre no cualificado, mientras que EXPORT hace público el símbolo.

Programa de ejemplo:

```
LOCAL square(x) // función local
```

```
BEGIN
```

```
RETURN x^2;
```

```
END;
```

```
helper(x) // función local
```

```
BEGIN
```

```
RETURN x+1;
```

```
END;
```

```
EXPORT demo() // función pública
```

```
BEGIN
```

```
LOCAL result;
```

```
result:=square(5)+helper(5);
```

```
RETURN result;
```

```
END;
```

1124 Logaritmo general

Sintaxis

LOG(valor [, base])

Función logaritmo general

Devuelve el logaritmo de valor en base. De forma predeterminada, base=10.

1129 Minúscula

ID 1129 · título recuperado de la página española

Sintaxis

LOWER(Cadena)
LOWER(Lista)

Para una cadena, devuelve la cadena con los caracteres en mayúsculas convertidos a minúsculas.

Para una lista de cadenas, devuelve la lista de cadenas con los caracteres en mayúsculas convertidos a minúsculas.

0649 Factorización LQ

ID 649 · título recuperado de la página española

Sintaxis

LQ(matriz)

Factoriza una matriz $m \times n$ en tres matrices: L, Q y P, donde L es una trapezoidal inferior $m \times n$, Q es una ortogonal $n \times n$ y P es una permutación $m \times m$; y $P \cdot A = L \cdot Q$.

0650 Mínimos cuadrados

ID 650 · título recuperado de la página española

Sintaxis

LSQ(matriz1, matriz2)

Devuelve la matriz de mínimos cuadrados de norma mínima (o vector) correspondiente al sistema $\text{matriz1} \cdot X = \text{matriz2}$

0651 Descomposición LU

ID 651 · título recuperado de la página española

Sintaxis

LU(A)

Factoriza una matriz cuadrada A en tres matrices L, U y P, donde L es una matriz triangular inferior, U una matriz triangular superior y P una matriz de permutación. Estas matrices satisfacen $P \cdot A = L \cdot U$.

0817 Funciones en la vista de Inicio (PPL): 'M'

ID 817 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'M'

0603 Crear lista

ID 603 · título recuperado de la página española

Sintaxis

MAKELIST(Expr, Var, Begin, End [, Incr])

Calcula una secuencia de elementos para una nueva lista.

Evalúa la expresión Expr, incrementando la variable Var desde los valores Begin hasta End, utilizando pasos de incremento Incr (el valor predeterminado es 1).

0620 Crear matriz

ID 620 · título recuperado de la página española

Sintaxis

MAKEMAT(Expr, Filas, Columnas)
MAKEMAT(Expr, Elementos)

Crea una matriz de Filas × Columnas de dimensión, con Expr para calcular cada elemento. Si Expr contiene las variables I y J, entonces el cálculo para cada elemento sustituye el número actual de la fila para I y el número de columna actual para J. También puede crear un vector utilizando el número de elementos en lugar del número de filas y columnas.

Ejemplos

[Ejemplo gráfico E1]
[Ejemplo gráfico E2]
[Ejemplo gráfico E3] solo en la Vista de Inicio
[Ejemplo gráfico E4] solo en la Vista de Inicio
[Ejemplo gráfico E5] solo en la Vista de Inicio

0535 Mantisa

ID 535 · título recuperado de la página española

Sintaxis

MANT(Valor)

Devuelve los dígitos significativos de Valor.

1736 MatToList — Convierte una matriz o un vector en una lista de listas o en una lista.

ID 1736 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: MatToList

Explicación española: Convierte una matriz o un vector en una lista de listas o en una lista.

Sintaxis

MatToList([])
MatToList([], [...])

Convierte una matriz o un vector en una lista de listas o en una lista.

0538 Máximo

ID 538 · título recuperado de la página española

Sintaxis

MAX(value1 [, value2, ..., value16])
MAX(list)
MAX(vector)

Devuelve el mayor de los valores reales dados o el mayor valor real en una lista o vector. Si se proporcionan múltiples listas, deben tener la misma longitud, y los elementos correspondientes en las mismas posiciones de esas listas se compararán. El resultado será una lista de la misma longitud que las listas de entrada.

1135 Real máximo

ID 1135 · título recuperado de la página española

Sintaxis

MAXREAL

Devuelve el número real más grande que HP Prime es capaz de representar en las Vistas de Inicio y CAS.

En la Vista de CAS, MAXREAL=1,79769313486E308

En Vista de Inicio, MAXREAL=9,99999999999E499

1138 Memoria del sistema

ID 1138 · título recuperado de la página española

Sintaxis

MEMORY
MEMORY(n)

Devuelve una lista con números enteros que representan la memoria y el espacio de almacenamiento, o el número entero individual en la posición n.

1140 Medio

ID 1140 · título recuperado de la página española

Sintaxis

MID(Cadena, Posición [, N])

Extrae n caracteres de la Cadena comenzando en la Posición. Si no se especifica N, entonces MID extrae el resto de la Cadena desde la Posición. Cada uno de los argumentos puede ser una lista, lo que permite un procesamiento rápido de un gran número de cadenas.

0539 Mínimo

ID 539 · título recuperado de la página española

Sintaxis

MIN(value1 [, value2, ..., value16])
MIN(list)
MIN(vector)

Devuelve el menor de los valores reales dados o el menor valor real en una lista o vector. Si se proporcionan múltiples listas, deben tener la misma longitud, y los elementos correspondientes en las mismas posiciones de esas listas se compararán. El resultado será una lista de la misma longitud que las listas de entrada.

1141 Real mínimo

ID 1141 · título recuperado de la página española

Sintaxis

MINREAL

Devuelve el número real más pequeño (cercano a cero) que HP Prime puede representar en las Vistas de Inicio y de CAS.

En la Vista de CAS, MINREAL=2,22507385851E-308

En la Vista de Inicio, MINREAL=1E-499

1142 Diferencia de dos listas o vectores

ID 1142 · título recuperado de la página española

Sintaxis

MINUS(Obj_1, Obj_2)

Toma dos argumentos Obj_1 y Obj_2, que pueden ser listas o vectores, y devuelve un vector que es el resultado de eliminar todos los elementos contenidos en Obj_2 de Obj_1.

1143 Convertir al sistema métrico

ID 1143 · título recuperado de la página española

Sintaxis

MKSA(Unidad_de_valor)

Convierte la medida Unidad_de_valor a su valor y unidad correspondiente en el equivalente MKSA de la unidad.

MKSA significa sistema de metros, kilogramos, segundos y amperios.

0540 Módulo

ID 540 · título recuperado de la página española

Sintaxis

valor1 MOD valor2

Devuelve el resto del valor1/valor2 de la división euclidiana.

1145 Evento Get Touch

ID 1145 · título recuperado de la página española

Sintaxis

MOUSE([index])

Devuelve una lista que contiene dos listas que describen la ubicación actual de cada posible puntero. Si un puntero no está activo, su lista está vacía. Cada lista de puntero no vacía tiene la forma {x, y, x original, y original, tipo}, donde tipo es 0 (para nuevo), 1 (para completado), 2 (para arrastrar), 3 (para estirar), 4 (para girar) y 5 (para clic largo).

Si no se ha producido ninguna actividad del puntero, MOUSE() devuelve {{}}, {{}}.

El parámetro opcional index está basado en cero y devuelve el elemento indexado de las dos listas de punteros como si se hubieran aplanado en orden. MOUSE(0) a MOUSE(4) devuelven x, y, x original, y original y tipo para el primer puntero. MOUSE(5) a MOUSE(9) devuelven x, y, x original, y original y tipo para el segundo puntero. Si no se ha producido actividad del puntero para el elemento de puntero solicitado, MOUSE(index) devuelve -1.

1147 Cuadro de mensaje

ID 1147 · título recuperado de la página española

Sintaxis

MSGBOX(expr [, OK_Cancel])
MSGBOX(cadena [, OK_Cancel])

Muestra un cuadro de mensaje con el valor de una expr o cadena.

Si OK_Cancel es verdadero, muestra las teclas de menú [[OK]] y [[Canc.]]; de lo contrario, muestra la tecla de menú [[OK]]. El valor predeterminado de OK_Cancel es falso.

Devuelve verdadero (no cero) si el usuario pulsa [[OK]], falso (0) si el usuario pulsa [[Canc.]].

0818 Funciones en la vista de Inicio (PPL): 'N'

ID 818 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'N'

0578 Densidad normal

ID 578 · título recuperado de la página española

Sintaxis

NORMALD([μ, σ,] x)

Función de densidad de probabilidad normal.

Calcula la densidad de probabilidad en el valor x, dados el promedio μ, y la desviación estándar σ, de una distribución normal.

Con un argumento, x, devuelve la densidad de probabilidad en x, asumiendo una media de cero y una desviación estándar de 1.

0586 Normal acumulada

Sintaxis

`NORMALD_CDF([μ, σ,] x [, x2])`

Función de distribución normal acumulada.

Con tres valores (μ , σ , y x), devuelve la probabilidad de cola inferior de la función de densidad de probabilidad normal para el valor x , dados el promedio μ , y la desviación estándar σ , de una distribución normal. Con el cuarto valor opcional $x2$, devuelve el área de la función de densidad de probabilidad normal entre los dos valores x .

Con un argumento x , devuelve la probabilidad de cola más baja de la función de densidad de probabilidad normal estándar para el valor x , suponiendo un promedio de cero y una desviación estándar de 1.

0594 Normal acumulada inversa**Sintaxis**

`NORMALD_ICDF([μ, σ,] p)`

Función de distribución normal acumulada inversa.

Devuelve el valor x de distribución normal acumulada asociado a la probabilidad de cola inferior p , dados el promedio μ , y la desviación estándar σ , de una distribución normal

Con un argumento, p , asume un promedio de cero y una desviación estándar de uno.

1158 NOT lógico**Sintaxis**

`NOT Valor`

Para números reales, devuelve 1 si Valor es un cero; de lo contrario, devuelve 0.

Para números enteros y cadenas, NOT se realiza a nivel de bit, cambiando todos los 1's a 0's y todos los 0's a 1's.

1159 Tecla de raíz n-ésima**Sintaxis**

`Valor1 NTHROOT Valor2`
`Valor1 √ Valor2`

NTHROOT: la función raíz n-ésima

Esta combinación de Shift-tecla proporciona una plantilla para la función NTHROOT. Devuelve la raíz principal Valor1 de Valor2. En el teclado, la función NTHROOT se representa por $\sqrt[n]{}$.

0819 Funciones en la vista de Inicio (PPL): 'O'

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'O'

1164 OR lógico**Sintaxis**

`Valor1 OR Valor2`

Para números reales, devuelve 1 si Valor1 o Valor2 no es un cero; de lo contrario, devuelve 0.

Para enteros y cadenas, OR se realiza a nivel de bit, y devuelve 1 si cada bit correspondiente es 1, de lo contrario, devuelve 0.

0820 Funciones en la vista de Inicio (PPL): 'P'

ID 820 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'P'

0570 Permutaciones

ID 570 · título recuperado de la página española

Sintaxis

PERM(*n*, *r*)

Devuelve el número de permutaciones (con respecto al orden) de *n* cosas considerando *r* por vez: $n!/(n-r)!$

1173 Función piecewise

ID 1173 · título recuperado de la página española

Sintaxis

PIECEWISE(*Test_1*, *Case_1* [, *Test_2*, *Case_2* , ..., *Test_N*, *Case_N*])

Se utiliza con la Entrada de la Configuración de Inicio configurada como Algebraica para introducir una función definida por la función a trozos (piecewise) en la Vista simbólica de la aplicación Función (entre otros usos). Toma como argumentos pares que constan de una condición que define un dominio de subfunción y una expresión que define la subfunción. Cada uno de estos pares define una subfunción de la función a trozos y el dominio en el que está activa.

Si se utiliza con la Entrada de la Configuración de Inicio configurada como Libro de texto o si se se accede a través del menú Plantilla, entonces la sintaxis varía ligeramente y se limita a dos trozos.

Sintaxis

```
{case1 si test1
...
{case8 [si test8]
```

Ejemplos

[Ejemplo gráfico E1] como $F_1(X)$ en la Vista simbólica de la aplicación Función aparece como:

X if $X < -4$

$F_1(X) = \{ X+1 \text{ if } -4 \leq X \text{ AND } X < 2$

$X+2 \text{ if } X \geq 2$

En la Vista de gráfico, se dibuja el gráfico de dos rayas y un segmento.

1286 Píxel Off

ID 1286 · título recuperado de la página española

Sintaxis

PIXOFF([*G*,] *x*, *y*)
PIXOFF([*G*,] *Coordinates*)

Establece uno o varios píxeles del GROB *G* en blanco mediante coordenadas cartesianas.

Parámetros

- *G*: GROB de destino.
- *x*, *y*: Coordenadas cartesianas de un único píxel.
- *Coordinates*: Coordenadas cartesianas proporcionadas como lista, vector o matriz.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1275 Píxel Off

ID 1275 · título recuperado de la página española

Sintaxis

```
PIXOFF_P([G,] x, y)  
PIXOFF_P([G,] Coordinates)
```

Establece uno o varios píxeles del GROB G en blanco mediante coordenadas de píxel.

Parámetros

- G: GROB de destino.
- x, y: Coordenadas de un único píxel.
- Coordinates: coordenadas de píxel proporcionadas como lista, vector o matriz.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1287 Píxel On

ID 1287 · título recuperado de la página española

Sintaxis

```
PIXON([G,] x, y [, Color])  
PIXON([G,] Coordinates [, Color])  
PIXON([G,] Coordinates, Colors)
```

Establece el color de uno o varios píxeles en el GROB G mediante coordenadas cartesianas. Si se especifica un único color, este se aplica a todos los píxeles especificados. Cuando se utiliza Coordinates con una lista de colores, cada color se aplica al píxel correspondiente.

Parámetros

- G: GROB de destino.
- x, y: Coordenadas cartesianas de un único píxel.
- Coordinates: Coordenadas cartesianas proporcionadas como lista, vector o matriz.
- Color: Color opcional aplicado a todos los píxeles especificados.
- Colors: Lista de colores aplicada a los píxeles correspondientes.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1276 Píxel On

ID 1276 · título recuperado de la página española

Sintaxis

```
PIXON_P([G,] x, y [, Color])  
PIXON_P([G,] Coordinates [, Color])  
PIXON_P([G,] Coordinates, Colors)
```

Establece el color de uno o varios píxeles en el GROB G mediante coordenadas de píxel. Si se especifica un único color, este se aplica a todos los píxeles especificados. Cuando se utiliza Coordinates con una lista de colores, cada color se aplica al píxel correspondiente.

Parámetros

- G: GROB de destino.
- x, y: Coordenadas de un único píxel.
- Coordinates: coordenadas de píxel proporcionadas como lista, vector o matriz.
- Color: Color opcional aplicado a todos los píxeles especificados.
- Colors: Lista de colores aplicada a los píxeles correspondientes.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

0584 Densidad de Poisson

ID 584 · título recuperado de la página española

Sintaxis

POISSON(μ , k)

Función de masa de probabilidad de Poisson

Calcula la probabilidad de que ocurran k incidencias de un evento en un intervalo futuro dado, el promedio de incidencias de tal evento en dicho intervalo en el pasado. Para esta función, k es un entero no negativo y μ es un número real.

1178 Coeficientes de polinomios

ID 1178 · título recuperado de la página española

Sintaxis

POLYCOEF(Vector)
POLYCOEF(Lista)

Devuelve los coeficientes del polinomio con las raíces especificadas en un vector o lista.

1177 Evaluación de polinomios

ID 1177 · título recuperado de la página española

Sintaxis

POLYEVAL(Vector, Valor)
POLYEVAL(Lista, Valor)

Dado un vector o lista de coeficientes y un valor, evalúa el polinomio dado por esos coeficientes en el valor determinado.

1245 forma polinómica

ID 1245 · título recuperado de la página española

Sintaxis

POLYFORM(Expr)

Devuelve la expresión expr en forma polinómica.

1176 Raíces de polinomios

ID 1176 · título recuperado de la página española

Sintaxis

POLYROOT(Poli)
POLYROOT(Vector)

Devuelve los ceros del polinomio dado como argumento (ya sea como una expresión simbólica o como un vector de coeficientes).

0607 Posición

ID 607 · título recuperado de la página española

Sintaxis

POS(lista, element)

Devuelve la posición del elemento dentro lista. Si hay más de una instancia de element, devuelve la posición de la primera aparición. Devuelve 0 si no hay ninguna ocurrencia del elemento especificado.

1301 Print

ID 1301 · título recuperado de la página española

Sintaxis

```
PRINT(expr)
PRINT(string)
PRINT()
```

Imprime el resultado de la expresión `expr` o la cadena `string` en el terminal. Las cadenas pueden contener la secuencia de escape `"\n"` para nuevas líneas.

El terminal es un mecanismo de visualización de salida de texto del programa que solo se muestra cuando se ejecutan comandos `PRINT`. Cuando está visible, puedes usar las teclas arriba/abajo para ver el texto, Retroceso para limpiar el terminal y cualquier otra tecla para ocultar el terminal.

Puedes mostrar el terminal en cualquier momento usando la combinación `On+T` (mantén presionada la tecla `On`, presiona la tecla `T`, luego suelta ambas teclas). Presionar `On` detiene la interacción con el terminal.

`PRINT` sin argumentos limpia el terminal.

1740 PRINT2D — Imprime el resultado de `expr` o una cadena en la terminal.

ID 1740 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `PRINT2D`

Explicación española: Imprime el resultado de `expr` o una cadena en la terminal.

Sintaxis

```
PRINT2D(obj, [tamaño_de_fuente], [color], banderas)
PRINT(2D)
```

Imprime el resultado de `expr` o una cadena en la terminal. La visualización se realizará utilizando la visualización de tipo manual.

La terminal es un mecanismo de visualización de salida de texto de programa que solo se muestra cuando se ejecutan comandos `PRINT`. Cuando es visible, puede usar las teclas de arriba/abajo para ver el texto, la tecla de retroceso para borrar el texto y cualquier otra tecla para ocultar la terminal.

Puede mostrar la terminal en cualquier momento usando la combinación `ON+T` (mantenga presionada la tecla `On`, presione la tecla `T` y luego suelte ambas teclas). Presionar `On` detiene la interacción con la terminal.

`PRINT` sin argumentos borra la terminal.

Puede especificar la fuente a utilizar, el color y la bandera le permite controlar algunos otros atributos a través del parámetro de bandera. El parámetro de bandera es un campo de bits donde `bit1` = negrita, `bit2` = subrayado, `bit3` = tachado, `bit4` = centrado y `bit5` = alineado a la derecha. Tenga en cuenta que negrita, subrayado y tachado solo están disponibles en cadenas, no en objetos 2D.

1734 PUT — Devuelve un nuevo objeto composite (lista, matriz, cadena o entero binario) en el que el objeto situado...

ID 1734 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `PUT`

Explicación española: Devuelve un nuevo objeto composite (lista, matriz, cadena o entero binario) en el que el objeto situado...

Sintaxis

```
PUT(composite, value, index1 [, ..., indexN])
```

Devuelve un nuevo objeto composite (lista, matriz, cadena o entero binario) en el que el objeto situado en el índice o los índices especificados por `index1` hasta `indexN` se sustituye por `value`.

Esto funciona de la misma manera que almacenar un valor en una variable indexada, pero permite trabajar con objetos que no están almacenados en una variable.

1485 PX→C — Transforma las coodernadas de píxel en coordenadas cartesianas.

ID 1485 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: PX→C

Explicación española: Transforma las coodernadas de píxel en coordenadas cartesianas.

Sintaxis

PX→C(x, y)

PX→C({x, y})

Transforma las coodernadas de píxel en coordenadas cartesianas. Devuelve una lista.

Ejemplos

[Ejemplo gráfico E1] (suponiendo que la Configuración de gráfico actual de la aplicación está configurada como predeterminada)

[Ejemplo gráfico E2] (suponiendo que la Configuración de gráfico actual de la aplicación está configurada como predeterminada)

1754 PYTHON

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de PYTHON.](#)

0821 Funciones en la vista de Inicio (PPL): 'Q'

ID 821 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'Q'

1723 Aproximación de real a cociente

ID 1723 · título recuperado de la página española

Sintaxis

QPI(expr [, d])

QPI intenta aproximar la expresión expr a una de las siguientes formas utilizando d dígitos de precisión:

- p/q
- $(a/b)*\sqrt{(p/q)}$
- $(p/q)*\pi$
- $\ln(p/q)$
- $e^{(p/q)}$

La expresión expr puede ser un número, un número complejo, una lista, un vector o una matriz.

0652 Factorización QR

ID 652 · título recuperado de la página española

Sintaxis

QR(matriz)

Factoriza una matriz m x n en tres matrices: {[[m x m ortogonal]], [[m x n trapezoidalsuperior]], [[n x n permutación]]}.

1332 QUOTE o ' '

ID 1332 · título recuperado de la página española

Sintaxis

QUOTE(expresión)
'expression'

Devuelve la expresión sin cambios y sin evaluar.

Esta función se utiliza principalmente con el comando STO► para almacenar una función en una variable.

Por ejemplo, si desea almacenar SIN(X) en F1, no puede introducir SIN(X)►F1 ya que SIN(X) se evaluaría y se almacenaría un resultado numérico en F1. En su lugar use, QUOTE(SIN(X))►F1 para almacenar SIN(X) en F1.

0822 Funciones en la vista de Inicio (PPL): 'R'

ID 822 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'R'

1219 Normal aleatorio

ID 1219 · título recuperado de la página española

Sintaxis

```
randnorm(mu, sigma)
```

Devuelve un número real aleatorio a partir de la distribución normal $N(\mu, \sigma)$.

0573 Entero aleatorio

ID 573 · título recuperado de la página española

Sintaxis de Inicio

```
RANDINT([Min=0,] Max=1]) RANDINT(Count, Min, Max)
```

Sintaxis CAS

```
randint([Min=1,] Max)
```

Devuelve un entero pseudoaleatorio generado utilizando un valor de semilla y actualiza el valor de semilla.

Parámetros

- Min, Max: Especifican el rango Min-Max para los enteros aleatorios.
- Count: El número de enteros a aleatorizar. Cuando se especifica, el resultado es una lista.

0622 Matriz aleatoria

ID 622 · título recuperado de la página española

Sintaxis

```
RANDMAT([Nombre_matriz,] Filas [, Columnas [, Entero]])  
RANDMAT([Nombre_matriz,] Filas [, Columnas [, Real1, Real2]])  
RANDMAT([Nombre_matriz,] Filas [, Columnas [, 'Función_generación']])
```

Crea una matriz aleatoria con el número especificado de filas y columnas. Si se proporciona Nombre_matriz (M0-M9), el resultado se almacenará allí.

Si no se proporcionan entradas adicionales, las entradas serán enteros que van de -99 a 99.

Si se proporciona un entero, las entradas serán enteros que van de 0 a ese Entero.

Si se proporcionan dos reales, las entradas serán reales de Real1 a Real2.

Si se proporciona la 'Función_generación', se utilizará para generar cada entrada en la matriz.

0574 Normal aleatorio

ID 574 · título recuperado de la página española

Sintaxis

```
RANDNORM([μ [, σ]])  
RANDNORM(n, μ, σ)
```

Devuelve un número aleatorio de distribución normal con el promedio μ especificado y la desviación estándar σ . Los valores predeterminados son 0 y 1.

Con tres argumentos, devuelve una lista de n tamaño donde cada elemento es un número aleatorio desde la distribución normal con el promedio μ y la desviación estándar σ especificados.

0572 Número aleatorio

ID 572 · título recuperado de la página española

Sintaxis

```
RANDOM([a [, b [, c]])
```

Devuelve un número pseudoaleatorio generado mediante un valor de inicialización y actualiza el valor de inicialización.

Sin un argumento, esta función devuelve un número x aleatorio con $0 \leq x < 1$.

Con un argumento, devuelve un número x aleatorio con $0 \leq x < a$.

Con dos argumentos, devuelve un número aleatorio con $a \leq x < b$.

Con tres argumentos, devuelve una lista de tamaño a , siendo cada elemento un número x aleatorio con $b \leq x < c$.

0576 Valor de inicialización aleatorio

ID 576 · título recuperado de la página española

Sintaxis

```
RANDSEED([valor])
```

Configura el valor de inicialización del generador de números aleatorios. Si no se introduce ningún valor, utiliza el valor de la hora actual como valor de inicialización.

0634 Clasificación de matriz rectangular

ID 634 · título recuperado de la página española

Sintaxis

```
RANK(matriz)
```

Proporciona una clasificación de matriz rectangular.

0547 Parte real

ID 547 · título recuperado de la página española

Sintaxis

```
RE(x+yi)
```

Devuelve la parte real de un número complejo.

1076 Leer una línea desde el terminal

ID 1076 · título recuperado de la página española

Sintaxis

```
READLINE()
```

Abre un terminal y espera a que el usuario ingrese una línea de texto. Cuando se presiona la tecla ENTER, devuelve el texto ingresado como una cadena. Si se presiona la tecla ESC o ON, devuelve una cadena vacía ("").

1288 Dibujar rectángulos

ID 1288 · título recuperado de la página española

Sintaxis

```
RECT([G,] [x1, y1 [, x2, y2]] [, fillColor])  
RECT([G,] x1, y1, x2, y2 [[, edgeColor], fillColor])
```

Dibuja un rectángulo en el GROB G mediante coordenadas cartesianas. Los puntos $(x1, y1)$ y $(x2, y2)$ definen las esquinas opuestas del rectángulo. El borde y el interior se dibujan utilizando los colores de borde y de relleno especificados.

Parámetros

- G: GROB de destino.
- x1, y1: Coordenadas cartesianas de la primera esquina. De forma predeterminada, se utiliza la esquina superior izquierda de G.
- x2, y2: Coordenadas cartesianas de la esquina opuesta. De forma predeterminada, se utiliza la esquina inferior derecha de G.
- fillColor: Color opcional del interior. Si se omite, se utiliza el color predeterminado.
- edgeColor: Color opcional del borde. Solo puede especificarse cuando también se proporciona fillColor. Si se omite, el borde utiliza el mismo color que el relleno.

Cuando se llama sin argumentos, RECT() borra la pantalla actual utilizando el color de relleno predeterminado. Cuando se llama únicamente con G, RECT(G) borra el GROB especificado de la misma manera.

Es posible dibujar bordes y rellenos semitransparentes mediante transparencia alfa. Para dibujar únicamente el borde, especifique un color de relleno completamente transparente.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1277 Rectángulo

ID 1277 · título recuperado de la página española

Sintaxis

```
RECT_P([G,] [x1, y1 [, x2, y2]] [, fillColor])  
RECT_P([G,] x1, y1, x2, y2 [, edgeColor], fillColor)
```

Dibuja un rectángulo en el GROB G mediante coordenadas de píxel. Los puntos (x1, y1) y (x2, y2) definen las esquinas opuestas del rectángulo. El borde y el interior se dibujan utilizando los colores de borde y de relleno especificados.

Parámetros

- G: GROB de destino.
- x1, y1: Coordenadas de la primera esquina. De forma predeterminada, se utiliza la esquina superior izquierda de G.
- x2, y2: Coordenadas de la esquina opuesta. De forma predeterminada, se utiliza la esquina inferior derecha de G.
- fillColor: Color opcional del interior. Si se omite, se utiliza el color predeterminado.
- edgeColor: Color opcional del borde. Solo puede especificarse cuando también se proporciona fillColor. Si se omite, el borde utiliza el mismo color que el relleno.

Cuando se llama sin argumentos, RECT_P() borra la pantalla actual utilizando el color de relleno predeterminado. Cuando se llama únicamente con G, RECT_P(G) borra el GROB especificado de la misma manera.

Es posible dibujar bordes y rellenos semitransparentes mediante transparencia alfa. Para dibujar únicamente el borde, especifique un color de relleno completamente transparente.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1291 Redimensionar

ID 1291 · título recuperado de la página española

Sintaxis

```
REDIM(nombrematriz, tamaño)
```

Cambia las dimensiones de la matriz o vector especificados al tamaño. Para una matriz, tamaño corresponde a una lista de dos enteros {n1, n2}. Para un vector, tamaño corresponde a una lista que contiene un entero {n}. Se conservan los valores existentes en la matriz. Los valores de relleno serán ceros.

1483 Reemplazar

ID 1483 · título recuperado de la página española

Sintaxis

REPLACE(Objeto1, Inicio, Objeto2)

Reemplaza parte de una matriz, vector o cadena (Objeto1) que comienza en Inicio con Objeto2.

Para una matriz, Inicio es una lista que contiene dos números; para un vector o cadena es un número único.

Nota: para las cadenas, puede hacer: REPLACE("cadena", "sub_cadena", "reemplazar_cadena")

1506 Repetir estructura de bucle

ID 1506 · título recuperado de la página española

Sintaxis

REPEAT comandos UNTIL prueba;

Executa comandos UNTIL prueba sea verdadera.

A:=5;

REPEAT

PRINT(A);

A:= A-1;

UNTIL A<1;

introduce 5 4 3 2 1

1730 Resource — La función Resource permite acceder a los recursos binarios almacenados como parte del programa desde el...

ID 1730 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Resource

Explicación española: La función Resource permite acceder a los recursos binarios almacenados como parte del programa desde el...

Sintaxis

Resource()

Resource(name)

Resource(name, pos)

Resource(name, start, end)

Resource(name):=data

Resource(name, pos):=data

La función Resource permite acceder a los recursos binarios almacenados como parte del programa desde el que se llama a Resource.

Los recursos binarios pueden contener datos binarios arbitrarios o imágenes almacenadas en formato PNG o JPEG.

Parámetros

- name: Nombre de un recurso binario existente, especificado como una cadena de texto.
- pos: Posición de un byte en el recurso. Al escribir una lista de bytes, es la posición en la que comienza la escritura.
- start: Posición del primer byte que se va a leer.
- end: Posición del último byte que se va a leer. Se incluye el byte situado en end.
- data: Datos que se van a escribir. Pueden ser un solo byte o una lista de valores de byte, cada uno de 0 a 255.

Creación y almacenamiento de recursos

No se puede crear un recurso binario desde un programa mediante la función Resource. Primero debe crearse desde el menú [[Más]] del Editor de programas, donde se especifican su nombre y su tipo.

Una vez creado, un recurso puede modificarse desde el Editor de programas o desde un programa mediante la función Resource.

Los recursos se almacenan de forma persistente como parte del programa, al igual que su código fuente. Los cambios realizados en un recurso se conservan después de finalizar el programa, tras ejecuciones y recompilaciones posteriores y después de restablecer la calculadora. El contenido modificado permanece almacenado hasta que el recurso vuelve a modificarse.

Aunque los recursos se almacenan de forma persistente con el programa, se cargan en la RAM junto con él y, por lo tanto, consumen memoria mientras el programa está cargado.

Resource solo funciona con recursos binarios. También se pueden crear recursos de código fuente desde el menú `[[Más]]`, pero Resource no los devuelve ni permite acceder a ellos. Consulte la variable Programs para obtener información sobre el acceso a los recursos de código fuente de un programa.

Listado de recursos

Resource() devuelve una lista que describe todos los recursos binarios asociados con el programa. Cada elemento de la lista devuelta contiene tres elementos:

{nombre, tipo, tamaño}

donde nombre es el nombre del recurso, tipo es un entero que identifica el tipo de recurso y tamaño es el tamaño del recurso en bytes.

Los tipos de recurso son:

- 0: Png image
- 1: Jpg image
- 2: Binary asset

Lectura de datos de recursos

Resource(name) devuelve el contenido completo del recurso especificado como una lista de valores de byte.

Resource(name, pos) devuelve el byte situado en la posición pos.

Resource(name, start, end) devuelve una lista que contiene los bytes desde start hasta end, ambos incluidos.

A diferencia de los bytes de los archivos, que se indexan a partir de 0, los bytes de los recursos se indexan a partir de 1. Por lo tanto, la posición 1 corresponde al primer byte de un recurso. Usar 0 como posición de un recurso devuelve Error: entrada no válida.

Escritura de datos de recursos

Resource(name):=data sustituye todo el contenido del recurso por los valores de byte de data. En esta forma, data es una lista de valores de byte y el tamaño del recurso se ajusta al tamaño de la lista.

Resource(name, pos):=data escribe data a partir de la posición pos.

Si data es un solo entero, se escribe un byte en la posición pos.

Si data es una lista de valores de byte, los valores se escriben consecutivamente a partir de la posición pos. Los bytes existentes en esas posiciones se sustituyen. Si los datos escritos se extienden más allá del final actual del recurso, el recurso se amplía según sea necesario.

El contenido completo de un recurso existente puede copiarse en otro recurso existente. Por ejemplo, si el programa contiene recursos llamados "source" y "target":

```
Resource("target"):=Resource("source")
```

copia todo el contenido de "source" en "target".

Carga de datos desde archivos

AFiles y AFilesB pueden utilizarse para cargar datos desde archivos en un recurso existente.

AFiles puede utilizarse para cargar un archivo de imagen en un recurso de imagen existente. Por ejemplo, si el programa contiene un recurso de tipo Png image llamado "mypng":

```
Resource("mypng"):=AFiles("image.png")
```

sustituye el contenido de "mypng" por la imagen cargada desde image.png.

Consulte AFiles y AFilesB para obtener más información.

Recursos de imagen

Los recursos de tipo Png image y Jpg image contienen datos PNG o JPEG codificados; no son objetos GROB.

Al crear o editar un recurso de imagen en el Editor de programas, el contenido actual de G1 puede copiarse al recurso desde el menú `[[Más]]`.

Para utilizar un recurso de imagen como gráfico en un programa, asigne el recurso completo a un GROB. Por ejemplo:

```
G2:=Resource("mypng")
```

A continuación, la imagen puede manipularse mediante las funciones GROB y de dibujo habituales.

1305 Comando Return

ID 1305 · título recuperado de la página española

Sintaxis

RETURN expresión;

Sale de una función y devuelve el valor de expresión (opcional).

Código de ejemplo:

```
EXPORT FACTORIAL(N)
BEGIN
IF N==1 THEN
RETURN 1;
ELSE
RETURN N*FACTORIAL(N-1);
END;
END;
```

0605 Invertir lista

ID 605 · título recuperado de la página española

Sintaxis de Inicio

REVERSE(List)

Sintaxis de CAS

reverse(String) reverse(List) reverse(Vector)

Si se proporciona una lista o un vector, devuelve una lista o un vector con sus elementos en orden inverso.

Para una cadena, devuelve la cadena con sus caracteres invertidos. Advertencia: los caracteres no ASCII pueden ser manejados incorrectamente en CAS.

1486 RGB — Devuelve un valor de color que se puede usar como parámetro de color de las funciones de dibujo.

ID 1486 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: RGB

Explicación española: Devuelve un valor de color que se puede usar como parámetro de color de las funciones de dibujo.

Sintaxis

RGB(R, G, B [, A])

Devuelve un valor de color que se puede usar como parámetro de color de las funciones de dibujo.

R, G y B especifican los componentes de color rojo, verde y azul, respectivamente, con valores de 0 a 255.

El parámetro opcional A especifica el componente alfa y va de 0 (opaco) a 255 (transparente).

1284 Parte derecha

ID 1284 · título recuperado de la página española

Sintaxis

RIGHT(Cadena, n)

Devuelve los últimos n caracteres de la cadena.

1283 Rotar

ID 1283 · título recuperado de la página española

Sintaxis

```
ROTATE(String, N)
ROTATE(G, Angle [, BackgroundColor])
ROTATE([TrgtG,] Angle, SrcG [, DestCenter])
ROTATE([TrgtG,] SrcG, DestPoint_1, DestPoint_2, DestPoint_3, DestPoint_4 [, SrcPoint_1, SrcPoint_2, SrcPoint_3, SrcPoint_4])
```

Gira el contenido de una cadena, gira un objeto gráfico o transforma una región de un GROB de origen sobre un área especificada de un GROB de destino.

La forma de cadena realiza una rotación circular de los caracteres de String. Las formas de rotación gráfica giran un GROB alrededor de su centro o dibujan un GROB de origen girado con su centro en un punto especificado. La forma de transformación asigna una región de un GROB de origen a un área de un GROB de destino definida por cuatro puntos de destino.

Parámetros

Parámetros

Rotación de una cadena

- String: Cadena cuyos caracteres se giran.
- N: Número de caracteres que se van a girar. Un valor positivo gira los caracteres hacia la izquierda y un valor negativo los gira hacia la derecha.

Rotación gráfica

- G: GROB que se va a girar alrededor de su centro.
- Angle: Ángulo de rotación.
- BackgroundColor: Color opcional utilizado para rellenar el área adicional creada al cambiar el tamaño del GROB G.
- TrgtG: GROB de destino.
- SrcG: GROB de origen.
- DestCenter: Punto de TrgtG en el que se coloca el centro del SrcG girado.

Transformación gráfica

- TrgtG: GROB de destino.
- SrcG: GROB de origen.
- DestPoint_1, DestPoint_2, DestPoint_3, DestPoint_4: Puntos que definen el área de destino sobre la que se asigna la región de origen.
- SrcPoint_1, SrcPoint_2, SrcPoint_3, SrcPoint_4: Puntos opcionales que definen la región de SrcG que se va a asignar.

Rotación de una cadena

ROTATE(String, N) realiza una rotación circular de los caracteres de String.

Si N es positivo, los primeros N caracteres se desplazan desde el principio de String hasta el final.

Si N es negativo, los últimos ABS(N) caracteres se desplazan desde el final de String hasta el principio.

Si ABS(N) es mayor que dim(String), String se devuelve sin cambios.

Rotación de un GROB alrededor de su centro

ROTATE(G, Angle [, BackgroundColor]) gira el GROB G alrededor de su centro. El tamaño de G se ajusta según sea necesario para contener la imagen girada y cualquier área adicional se rellena con BackgroundColor.

Dibujo y posicionamiento de un GROB girado

ROTATE([TrgtG,] Angle, SrcG [, DestCenter]) dibuja SrcG, girado según Angle, en TrgtG. El centro del SrcG girado se coloca en DestCenter.

DestCenter se expresa en coordenadas de píxel. Si se omite DestCenter, se utiliza el centro de TrgtG.

Transformación de una región de origen

ROTATE([TrgtG,] SrcG, DestPoint_1, DestPoint_2, DestPoint_3, DestPoint_4 [, SrcPoint_1, SrcPoint_2, SrcPoint_3, SrcPoint_4]) asigna una región de SrcG al área de TrgtG definida por los cuatro puntos de destino.

Los puntos de origen y de destino se expresan en coordenadas de píxel y se corresponden según su índice.

Si se omiten los puntos de origen, se utilizan de forma predeterminada las esquinas de SrcG en el orden siguiente:

- SrcPoint_1: Esquina superior izquierda.
- SrcPoint_2: Esquina superior derecha.
- SrcPoint_3: Esquina inferior derecha.

- SrcPoint_4: Esquina inferior izquierda.

La asignación se realiza internamente mediante dos triángulos. Los puntos 1, 2 y 3 forman el primer triángulo, y los puntos 1, 3 y 4 forman el segundo. Como los triángulos se transforman de forma independiente, las configuraciones de puntos no uniformes pueden producir un estiramiento diferente en cada triángulo.

Un conjunto de puntos puede definir un triángulo especificando la misma posición para el primer y el segundo punto.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

0533 Redondear

ID 533 · título recuperado de la página española

Sintaxis

ROUND(valor [, lugares])

Redondea valor a la configuración de la pantalla del sistema. Si se dan lugares opcional, redondea valor a lugares decimales. Si lugares es negativo, redondea a dígitos significativos en su lugar.

0629 Norma de fila

ID 629 · título recuperado de la página española

Sintaxis

ROWNORM(matriz)

Busca el valor máximo (en todas las filas) para las sumas de los valores absolutos de todos los elementos de una fila.

0618 Matriz escalonada reducida

ID 618 · título recuperado de la página española

Sintaxis

RREF(matriz)

Cambia una matriz rectangular a su forma escalonada reducida.

1297 Real a base

ID 1297 · título recuperado de la página española

Sintaxis

R→B(real [, bits [, base]])

Convierte un número real en un entero binario.

Si el valor de entrada tiene una parte fraccionaria, se redondea al entero más cercano antes de la conversión. Los parámetros opcionales bits y base especifican la representación del entero y el sistema de numeración utilizado para mostrar el resultado.

Parámetros

- real: El número real que se va a convertir. Si tiene una parte fraccionaria, se redondea al entero más cercano antes de la conversión.
- bits: Parámetro opcional que especifica el ancho en bits y si el entero resultante tiene signo o no.
 $1 \leq \text{bits} \leq 64$ — Entero sin signo
 $-63 \leq \text{bits} \leq -1$ — Entero con signo
- base: Parámetro opcional que especifica el sistema de numeración utilizado para mostrar el entero binario resultante.

0: Configuración actual del sistema

1: Binario

2: Octal

3: Decimal

4: Hexadecimal

Ejemplos

[Ejemplo gráfico E1] (Si la base del sistema es hexadecimal)
[Ejemplo gráfico E2]
[Ejemplo gráfico E3]

0823 Funciones en la vista de Inicio (PPL): 'S'

ID 823 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'S'

1296 SCALE — Multiplica una fila de una matriz por un escalar.

ID 1296 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: SCALE

Explicación española: Multiplica una fila de una matriz por un escalar.

Sintaxis

SCALE(matrixname, value, row_number)

Multiplica una fila de una matriz por un escalar. La matriz se especifica mediante matrixname, el multiplicador escalar es value, y la fila a multiplicar es row_number.

1292 SCALEADD — Multiplica la fila1 especificada del nombre de la matriz por valor.

ID 1292 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: SCALEADD

Explicación española: Multiplica la fila1 especificada del nombre de la matriz por valor.

Sintaxis

SCALEADD(nombrematriz, valor, fila1, fila2)

Multiplica la fila1 especificada del nombre de la matriz por valor. A continuación, agrega este resultado a la segunda fila2 especificada de la matriz nombrematriz.

0653 Descomposición Schur

ID 653 · título recuperado de la página española

Sintaxis

SCHUR(matriz)

Factoriza una matriz cuadrada en dos matrices.

Si la matriz es real, el resultado es {[[ortogonal]], [[triangular cuasi superior]]}.

Si el modo complejo está activado y la matriz es compleja, el resultado es {[[unitario]], [[triangular superior]]}.

0556 Secante

ID 556 · título recuperado de la página española

Sintaxis

SEC(valor)

Secante: 1/COS(X).

Ejemplos

[Ejemplo gráfico E1] (en modo grados)
[Ejemplo gráfico E2]
[Ejemplo gráfico E3] (en modo grados)
[Ejemplo gráfico E4]

0548 Vector de la unidad o signo

ID 548 · título recuperado de la página española

Sintaxis

SIGN(valor)
SIGN(x+y□)

Devuelve el signo de valor.

Si es positivo, el resultado es 1; Si es negativo, -1. Si es cero, el resultado es cero.

Para valores complejos introducidos devuelve el vector de la unidad.

1520 SERIAL — Devuelve el número de serie de la calculadora

ID 1520 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: SERIAL
Explicación española: Devuelve el número de serie de la calculadora

Sintaxis

SERIAL

Devuelve el número de serie de la calculadora

1203 Elementos únicos

ID 1203 · título recuperado de la página española

Sintaxis

set(List)
set(Vector)

Devuelve un vector de elementos únicos de la lista o vector proporcionado (elimina duplicados).

1298 Configurar Base

ID 1298 · título recuperado de la página española

Sintaxis

SETBASE(#entero[m] [, C])

Muestra entero expresado en base m en base de lo que se indica mediante C.

El marcador base m puede ser b (para binario), d (para decimal), o (para octal), d (para decimal), o h (para hexadecimal). Si se omite m, se asume que el valor introducido está en la base predeterminada.

C = 0 Sistema

C = 1 Binario

C = 2 Octal

C = 3 Decimal

C = 4 Hexadecimal

Si se omite C, el resultado se muestra en la base predeterminada.

Ejemplos

[Ejemplo gráfico E1]
[Ejemplo gráfico E2] (si la base predeterminada es hexadecimal)
[Ejemplo gráfico E3]

1299 Configurar Bits

ID 1299 · título recuperado de la página española

Sintaxis

SETBITS(#entero[m] [, bits])

Establece el número de bits para representar entero.

El valor de bits debe estar en el rango de -63 a 64. El marcador de base m puede ser b (para binario), d (para decimal), o (para octal), d (para decimal) o h (para hexadecimal). Si se omite el marcador de base m o bits, se utiliza el valor predeterminado.

1274 Seno

ID 1274 · título recuperado de la página española

Sintaxis

SIN(Valor)

Devuelve el seno de Valor.

El valor se interpreta como radianes, grados o gradientes, según la configuración de la medición de ángulo en la Configuración de inicio, Configuración CAS o Configuración simbólica.

Ejemplos

[Ejemplo gráfico E1] (en modo grados)
[Ejemplo gráfico E2]
[Ejemplo gráfico E3] (en modo grados)
[Ejemplo gráfico E4]

0561 Seno hiperbólico

ID 561 · título recuperado de la página española

Sintaxis

SINH(valor)

Seno hiperbólico

0608 Tamaño de la lista

ID 608 · título recuperado de la página española

Sintaxis

SIZE(lista)

Devuelve la cantidad de elementos de una lista. Con una matriz, devuelve las dimensiones de la matriz.

0604 Ordenar lista

ID 604 · título recuperado de la página española

Sintaxis

SORT(List [, Sort_by])

Ordena los elementos de una lista en orden ascendente. Para listas compuestas (listas de listas o listas que contienen cadenas), Sort_by identifica el número de elemento dentro de cada objeto en la lista que se utilizará para ordenar.

La lista debe estar en el formato {object_1, object_2, ..., object_n}.

Todos los objetos ordenados deben ser del mismo tipo (por ejemplo, números, cadenas, listas...).

Si se están ordenando listas enteras (las listas enteras se ordenan como objetos únicos, pero su orden interno permanece sin cambios), todos los elementos en la posición especificada por Sort_by en las listas deben ser del mismo tipo, ya que estos elementos se comparan para fines de ordenación.

Ejemplos

[Ejemplo gráfico E1]
[Ejemplo gráfico E2]
[Ejemplo gráfico E3] (ordenar la lista por el 2º elemento de cada objeto)
[Ejemplo gráfico E4] (ordenar la lista por el 3er elemento de cada objeto)
[Ejemplo gráfico E5] (ordenar cada lista por el 2º elemento del 3er objeto)
[Ejemplo gráfico E6] (ordenar la lista por el 3er elemento seguido del 1er elemento de cada objeto)

0632 Radio espectral

ID 632 · título recuperado de la página española

Sintaxis

SPECRAD(matriz)

Devuelve el radio espectral de una matriz cuadrada.

0631 Norma espectral

ID 631 · título recuperado de la página española

Sintaxis

SPECNORM(matriz)

Devuelve la norma espectral de una matriz cuadrada.

1430 Raíz cuadrada

ID 1430 · título recuperado de la página española

Sintaxis

SQRT(Expr)
 $\sqrt{\text{Expr}}$

Devuelve la raíz cuadrada positiva de Expr o el resultado numérico de Expr. También acepta listas.

1495 Inicia la aplicación

ID 1495 · título recuperado de la página española

Sintaxis

STARTAPP("NombreApl")

Inicia la aplicación NombreApl. La función START de la aplicación se ejecutará si está presente. Se iniciará la vista predeterminada de la aplicación. Tenga en cuenta que la función START se ejecuta siempre que el usuario pulsa la tecla de menú [[INICIO]] en la biblioteca de aplicaciones. También funciona con las aplicaciones guardadas en la biblioteca de aplicaciones.

1300 Iniciar Vista

ID 1300 · título recuperado de la página española

Sintaxis

STARTVIEW(ViewNumber [, Redraw])

Abre en la aplicación actual la vista especificada por ViewNumber. Redraw es opcional. Si Redraw es verdadero (distinto de 0), se actualiza la vista.

Los siguientes valores se pueden utilizar para ViewNumber:

- 0: Simbólico
- 1: Gráfico
- 2: Numérica
- 3: Configuración simbólica
- 4: Configuración de gráfico

- 5: Configuración numérica
- 6: Información de la aplicación
- 7: Menú Vistas

Si la aplicación actual tiene vistas definidas en el menú Vistas, se pueden utilizar los siguientes valores adicionales para ViewNumber:

- 8: Primera vista especial (P. divid.: det. de gráf.)
- 9: Segunda vista especial (P. div.: tabla de gráf.)
- 10: Tercera vista especial (Escala automática)
- 11: Cuarta vista especial (Decimal)
- 12: Quinta vista especial (Entero)
- 13: Sexta vista especial (Trig.)

Si ViewNumber es negativo, selecciona una de las siguientes vistas globales:

- -1: Pantalla de Inicio
- -2: Modos
- -3: Gestor de memoria
- -4: Biblioteca de aplicaciones
- -5: Catálogo de matrices
- -6: Catálogo de listas
- -7: Catálogo de programas
- -8: Catálogo de notas

1285 STRING — Evalúa Expresión y devuelve el resultado como una cadena.

ID 1285 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: STRING

Explicación española: Evalúa Expresión y devuelve el resultado como una cadena.

Sintaxis

STRING(Expresión, [Modo], [Precisión], [Separador o {Separador, ["[Puntodecimal[Exponente[SignoNegativo]]"]], [PuntoZero]]}, [LimitTam o {LimitTam, [TamFuente], [Negrita], [Itálica], [SpacioÚnico]]})

Evalúa Expresión y devuelve el resultado como una cadena.

Los parámetros Modo, Precisión y Separador especifican cómo se muestran los números.

Si se especifica Modo, es:

- 0: usar configuración actual
- 1: estándar
- 2: fijo
- 3: científico
- 4: Ingeniería
- 5: flotante
- 6: redondeo

Añadir 7 a este valor para especificar el modo de fracción propia y 14 para el modo de fracción mixta.

Precisión es -1 para la configuración actual o 0 a 12.

Separador puede ser un número. -1 significa uso predeterminado, 0 a 10 corresponden a las 11 opciones de agrupación de dígitos incorporadas disponibles en la configuración de inicio.

OR

Separador puede ser una cadena que contiene un conjunto de dígitos y separadores. Se supone que el último dígito se encuentra justo antes del punto decimal.

"[Puntodecimal[Exponente[SignoNegativo]]]" es una cadena de 0 a 3 caracteres. La primera de ellas se utilizará para la coma decimal, la segunda para el exponente y el último para el signo negativo.

Si PuntoZero no es un cero, entonces los números entre -1 y 1 se muestran sin un cero a la izquierda (por ejemplo, .1 en vez de 0.1)

Si se especifica LimitTam, el comando intentará generar una cadena que se ajuste al número de píxeles especificados. TamFuente se utiliza con Negrita, Itálica y SpacioÚnico (si su valor es distinto de cero) para estimar

la longitud máxima de la cadena que encajará.

Los valores para TamFuente son:

0=fuente actual (valor predeterminado)

1=font_10

2=font_12 (Pequeña)

3=font_14 (Mediana)

4=font_16 (Grande)

5=font_18

6=font_20

7=font_22

Ejemplos

[Ejemplo gráfico E1] cuando el formato actual es Estándar.

[Ejemplo gráfico E2] usando formato Fijo, precisión 4.

[Ejemplo gráfico E3] usando formato Fijo, precisión actual.

[Ejemplo gráfico E4] usando formato Fijo, precisión 3.

[Ejemplo gráfico E5] usando formato Fijo, precisión 3, agrupación personalizada.

[Ejemplo gráfico E6] usando formato Fijo, precisión 3, agrupación personalizada, separador decimal ";".

[Ejemplo gráfico E7] ajustado a 40 píxeles usando fuente 5.

1282 Cadena del identificador

ID 1282 · título recuperado de la página española

Sintaxis

STRINGFROMID(IdentificadorDeCadena [, Idioma])

Devuelve la cadena incorporada asociada con el identificador de cadena IdentificadorDeCadena. Usa el idioma actual si no se especifica el Idioma. Al momento de escribir, hay 3767 cadenas organizadas en 41 grupos. La mayoría de las cadenas tienen traducciones en 8 idiomas.

Para recuperar una lista de cadenas de un grupo específico, use STRINGFROMID(-n), que para un número de grupo dado n (proporcionado como un entero negativo), devuelve una lista de cadenas donde el primer elemento es el nombre del grupo y los elementos posteriores son los nombres de las cadenas dentro de ese grupo. Para crear un IdentificadorDeCadena completo, concatene el nombre del grupo con cualquiera de estos nombres de cadena, separados por un carácter "_" (guion bajo) (ver ejemplos).

Tanto IdentificadorDeCadena como Idioma pueden ser listas, lo que permite una sola llamada para recuperar múltiples cadenas de un grupo dado (en un idioma específico) o traducciones en varios idiomas para un solo identificador de cadena.

Considere la siguiente línea de comando:

```
L2:=STRINGFROMID(-1); STRINGFROMID(L2(1)+"_"+SUB(L2,2,SIZE(L2)),3)
```

La primera parte almacenará en L2 el nombre del grupo para el grupo 1 ("Main") y todos los nombres de cadenas dentro de este grupo. La segunda parte concatena el nombre del grupo con cada nombre de cadena extraído de la lista y recupera sus traducciones en francés.

La forma antigua de STRINGFROMID(Entero [, Idioma]) ahora está obsoleta y no debe usarse.

0579 Densidad T de Student

ID 579 · título recuperado de la página española

Sintaxis

STUDENT(d, x)

Función de densidad de probabilidad t de Student

Calcula la densidad de probabilidad de distribución t de Student en x, con d grados de libertad.

1293 Extraer parte

ID 1293 · título recuperado de la página española

Sintaxis

SUB(objeto, inicio, final)

Extrae una parte de una lista, un vector o una matriz.

Para una matriz, `inicio` y `final` son dos listas de dos números (`{fila, col}`) que especifican la esquina superior izquierda y la esquina inferior derecha de la porción a extraer.

Para un vector o una lista, `inicio` y `final` son dos números que especifican los índices del primer y el último elemento de la porción a extraer.

Nota: En CAS, SUB modifica el objeto directamente. Si necesitas conservar el original sin cambios, haz una copia antes de llamar a SUB.

En la vista de inicio, la variable original permanece sin cambios.

1289 Copiar GROB en el Destino

ID 1289 · título recuperado de la página española

Sintaxis

```
SUBGROB([srcG,] [x1, y1 [, x2, y2],] trgtG)
```

Crea o sustituye el GROB de destino `trgtG` por una copia de una región rectangular del GROB de origen `srcG` mediante coordenadas cartesianas. Los puntos `(x1, y1)` y `(x2, y2)` definen las esquinas opuestas de la región copiada. No es necesario inicializar previamente el GROB de destino.

Parámetros

- `srcG`: GROB de origen.
- `x1, y1`: Coordenadas cartesianas de la primera esquina. De forma predeterminada, se utiliza la esquina superior izquierda de `srcG`.
- `x2, y2`: Coordenadas cartesianas de la esquina opuesta. De forma predeterminada, se utiliza la esquina inferior derecha de `srcG`.
- `trgtG`: GROB de destino. Puede utilizarse cualquier GROB excepto `G0`.

Si no se especifican coordenadas, se utiliza todo el GROB de origen.

Por ejemplo, `SUBGROB(G1, G4)` crea o sustituye `G4` por una copia de todo el contenido de `G1`.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1278 Copiar GROB en el Destino

ID 1278 · título recuperado de la página española

Sintaxis

```
SUBGROB_P([srcG,] [x1, y1 [, x2, y2],] trgtG)
```

Crea o sustituye el GROB `trgtG` por una copia de una región rectangular del GROB de origen `srcG` mediante coordenadas de píxel. Los puntos `(x1, y1)` y `(x2, y2)` definen las esquinas opuestas de la región copiada. No es necesario inicializar previamente el GROB de destino.

Parámetros

- `srcG`: GROB de origen.
- `x1, y1`: Coordenadas de la primera esquina. De forma predeterminada, se utiliza la esquina superior izquierda de `srcG`.
- `x2, y2`: Coordenadas de la esquina opuesta. De forma predeterminada, se utiliza la esquina inferior derecha de `srcG`.
- `trgtG`: GROB de destino. Puede utilizarse cualquier GROB excepto `G0`.

Si no se especifican coordenadas, se utiliza todo el GROB de origen.

Por ejemplo, `SUBGROB_P(G1, G4)` crea o sustituye `G4` por una copia de todo el contenido de `G1`.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1648 Eliminar elementos

ID 1648 · título recuperado de la página española

Sintaxis

```
SUPPRESS(objeto, Índice)
SUPPRESS(objeto, inicio, final)
SUPPRESS(objeto, {índice1, índice2, ... ÍndiceN})
SUPPRESS(cadena1, cadena2)
```

Elimina elementos de objeto utilizando una sola posición Índice, un rango de índice inicio y final, o una lista de índices. objeto puede ser una lista, un vector o una cadena.

En el caso de cadena1 y cadena2, cada instancia de cada carácter en cadena2 será eliminado de cadena1.

0655 Valores únicos

ID 655 · título recuperado de la página española

Sintaxis

```
SVL(matriz)
```

Devuelve un vector que contiene los valores singulares de matriz.

1294 Intercambiar columnas

ID 1294 · título recuperado de la página española

Sintaxis

```
SWAPCOL(nombrematriz, columna1, columna2)
```

Intercambia columna1 y columna2 en la matriz especificada nombrematriz.

1295 Intercambiar filas

ID 1295 · título recuperado de la página española

Sintaxis

```
SWAPROW(nombrematriz, fila1, fila2)
```

Intercambia fila1 y fila2 en la matriz especificada nombrematriz.

0824 Funciones en la vista de Inicio (PPL): 'T'

ID 824 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'T'

1414 Tangente

ID 1414 · título recuperado de la página española

Sintaxis

```
TAN(Valor)
```

Devuelve la tangente de Valor.

El valor se interpreta como radianes, grados o gradientes, según la configuración de la medición de ángulo en la Configuración de inicio, Configuración CAS o Configuración simbólica.

Ejemplos

```
[Ejemplo gráfico E1] (en modo grados)
[Ejemplo gráfico E2]
[Ejemplo gráfico E3] (en modo grados)
[Ejemplo gráfico E4]
```

0565 Tangente hiperbólica

ID 565 · título recuperado de la página española

Sintaxis

TANH(valor)

Tangente hiperbólica

1741 Terminal — La función Terminal te da acceso a los datos del terminal y te permite controlarlos...

ID 1741 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Terminal

Explicación española: La función Terminal te da acceso a los datos del terminal y te permite controlarlos...

Sintaxis

```
Terminal
Terminal(n)
Terminal(-n)
Terminal(0)
Terminal:={ }
Terminal(n):= obj
Terminal(-n):= {obj, color, fuente, banderas}
Terminal(0):= {visible, maxLineas, xpos, ypos}
```

La función Terminal te da acceso a los datos del terminal y te permite controlarlos...

Terminal devuelve una lista de todos los elementos actualmente mostrados por el terminal.

Terminal(n) te permite acceder al enésimo elemento mostrado por el terminal.

Terminal(-n) devuelve el enésimo elemento y los metadatos asociados como una lista. El primer elemento es el elemento mostrado. Los otros 3 elementos son el color, la fuente y las banderas que siguen a la información de la función PRINT2D.

Terminal(0) devuelve una lista de varias informaciones del Terminal: {terminal_visible, max_Numero_De_Lineas_Visibles, {Current_xscroll, max_xscroll}, {current_yscroll, max_yscroll}, nb_Lineas_en_el_terminal}.

Terminal := {objetos} permite cambiar el objeto mostrado por el terminal de una vez.

Terminal(n) := obj permite cambiar el enésimo objeto mostrado por el terminal.

Terminal(-n) := {obj, color, fuente, banderas} permite cambiar el enésimo objeto mostrado por el terminal, incluidos los metadatos.

Terminal(0) := {visible, maxLineas, xpos, ypos} te permite controlar los parámetros del terminal, la visibilidad, el número máximo de líneas y las posiciones de desplazamiento. Ten en cuenta que un -1 en cualquiera de estos significa "sin cambios".

1302 Evaluación del tiempo

ID 1302 · título recuperado de la página española

Sintaxis

TEVAL(Param)

Devuelve el tiempo que se tarda en evaluar el parámetro.

Ejemplos

[Ejemplo gráfico E1] (Nota: el resultado real variará, pero deberá ser cercano a 5.00_s)

1290 Dibujar texto

ID 1290 · título recuperado de la página española

Sintaxis

TEXTOUT(Text [, G], x, y [, Font [, TextColor [, Width [, BackgroundColor]]]])

Dibuja texto en el GROB G mediante coordenadas cartesianas, comenzando en la posición (x, y), con una fuente, un color de texto, una anchura máxima y un color de fondo opcionales.

Parámetros

- Text: Texto que se va a dibujar.
- G: GROB de destino.

- x, y: Coordenadas cartesianas de la posición inicial.
- Font: Tamaño de fuente opcional.
- TextColor: Color opcional del texto.
- Width: Anchura máxima opcional del texto representado, en unidades cartesianas.
- BackgroundColor: Color opcional utilizado para el fondo situado detrás del texto representado. Si se omite, los píxeles existentes no se modifican.

Los tamaños de fuente disponibles son:

- 0: fuente predeterminada (determinada por la configuración actual del sistema)
- 1: fuente 10
- 2: fuente 12 (Pequeña)
- 3: fuente 14 (Mediana)
- 4: fuente 16 (Grande)
- 5: fuente 18
- 6: fuente 20
- 7: fuente 22

Devuelve la coordenada horizontal cartesiana en la que se dibujaría el siguiente carácter si la cadena contuviera caracteres adicionales.

TEXTOUT dibuja texto hasta una anchura máxima de 511 píxeles. El texto que exceda este límite se trunca.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1279 Dibujar texto

ID 1279 · título recuperado de la página española

Sintaxis

```
TEXTOUT_P(Text [, G], x, y [, Font [, TextColor [, Width [, BackgroundColor]]]])
```

Dibuja texto en el GROB G mediante coordenadas de píxel, comenzando en la posición (x, y), con una fuente, un color de texto, una anchura máxima y un color de fondo opcionales.

Parámetros

- Text: Texto que se va a dibujar.
- G: GROB de destino.
- x, y: coordenadas de píxel de la posición inicial.
- Font: Tamaño de fuente opcional.
- TextColor: Color opcional del texto.
- Width: Anchura máxima opcional del texto representado, en píxeles.
- BackgroundColor: Color opcional utilizado para el fondo situado detrás del texto representado. Si se omite, los píxeles existentes no se modifican.

Los tamaños de fuente disponibles son:

- 0: fuente predeterminada (determinada por la configuración actual del sistema)
- 1: fuente 10
- 2: fuente 12 (Pequeña)
- 3: fuente 14 (Mediana)
- 4: fuente 16 (Grande)
- 5: fuente 18
- 6: fuente 20
- 7: fuente 22

Devuelve la coordenada horizontal de píxel en la que se dibujaría el siguiente carácter si la cadena contuviera caracteres adicionales.

TEXTOUT_P dibuja texto hasta una anchura máxima de 511 píxeles. El texto que exceda este límite se trunca.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1280 Medir el tamaño del texto

ID 1280 · título recuperado de la página española

Sintaxis

TEXTSIZE(text [, font])

Devuelve una lista {ancho, alto} que contiene las dimensiones, en píxeles, del texto especificado cuando se representa con la fuente seleccionada.

Parámetros

- text: Texto que se va a medir.
- font: Tamaño de fuente opcional de 0 a 7. Si se omite, se utiliza 0. El valor 0 utiliza la fuente predeterminada determinada por la configuración actual del sistema.

1303 Valor de ticks internos

ID 1303 · título recuperado de la página española

Sintaxis

TICKS()

Devuelve el valor del reloj interno en milisegundos.

1391 Variable System Time

ID 1391 · título recuperado de la página española

Devuelve la hora del sistema en formato DMS. El formato DMS es HH°MM'SS" donde HH = hora, MM = minutos y SS = segundos.

Time:= HH°MM'SS" establece la hora.

0636 Trazar

ID 636 · título recuperado de la página española

Sintaxis

TRACE(matriz)

Busca la traza de una matriz cuadrada, igual a la suma de los elementos diagonales (igual también a la suma de los valores Eigen).

1489 Dibujar triángulo

ID 1489 · título recuperado de la página española

Sintaxis

```
TRIANGLE([G,] x1, y1, x2, y2, x3, y3 [, c [, alpha]])
TRIANGLE([G,] x1, y1, x2, y2, x3, y3, c1 [, c2, c3], alpha)
TRIANGLE([G,] x1, y1, x2, y2, x3, y3, c1, [c2, c3 [, alpha [, zstring, z1, z2, z3]]])
TRIANGLE([G,] {x1, y1 [, c1 [, z1]]}, {x2, y2 [, c2 [, z2]]}, {x3, y3 [, c3 [, z3]]} [, zstring])
TRIANGLE([G,] points_definition, triangle_definitions, rotation_matrix or {rotation_matrix or -1, ["N"], [{eye_x, eye_y, eye_z} or -1], [{xmin3D, xmax3D, ymin3D, ymax3D, zmin3D, zmax3D}]}, [zstring])
TRIANGLE([G,] pre_rotated_points, triangle_definitions [, zstring])
TRIANGLE([G,])
```

Dibuja uno o varios triángulos rellenos en el GROB G mediante coordenadas cartesianas. Un triángulo puede dibujarse con un único color o con colores independientes para sus vértices. Cuando se especifican tres colores de vértice, estos se interpolan sobre la superficie del triángulo. La transparencia se controla exclusivamente mediante el parámetro independiente alpha; el componente alfa de los valores de color se ignora.

Las formas avanzadas permiten renderizar varios triángulos en una sola operación y aplicar opcionalmente transformaciones geométricas, proyección en perspectiva, recorte 3D, normalización de profundidad y recorte de profundidad por píxel.

Parámetros

- G: GROB de destino.

- x1, y1: Coordenadas cartesianas del primer vértice.
- x2, y2: Coordenadas cartesianas del segundo vértice.
- x3, y3: Coordenadas cartesianas del tercer vértice.
- c: Color opcional aplicado a todo el triángulo.
- c1, c2, c3: Colores asignados a los vértices correspondientes. Si se especifican los tres colores, estos se interpolan sobre la superficie del triángulo.
- alpha: Transparencia opcional aplicada a todo el triángulo, de 0 (completamente opaco) a 255 (completamente transparente).
- z1, z2, z3: Valores de profundidad asignados a los vértices correspondientes. Cuando se utilizan con zstring, estos valores deben estar comprendidos entre 0 y 255.
- zstring: Cadena opcional de búfer de profundidad utilizada para el recorte de profundidad por píxel. Debe crearse mediante TRIANGLE([G]) y se actualiza a medida que se dibujan los triángulos.

Renderizado de varios triángulos

- points_definition: Lista o matriz que contiene los puntos utilizados por los triángulos. Cada definición de punto contiene las coordenadas x e y y puede incluir además una coordenada z, un color o ambos. Las coordenadas también pueden proporcionarse como un único valor de punto.
- triangle_definitions: Lista o matriz que describe los triángulos que se van a renderizar. Cada definición de triángulo contiene entre tres y cinco valores con la forma p1, p2, p3 [, c [, alpha]].
- p1, p2, p3: Índices de las tres entradas de points_definition que definen el triángulo.
- c: Color opcional que sustituye los colores almacenados en las definiciones de punto correspondientes. Para especificar alpha sin sustituir los colores de los puntos, utilice -1 para c.

Varias definiciones de triángulo pueden compartir el mismo color y transparencia mediante la forma agrupada {c [, alpha], triangle_1, ..., triangle_n}.

Transformación y proyección

- rotation_matrix: Matriz de transformación con dimensiones comprendidas entre 2x2 y 3x4. Cada punto se multiplica por esta matriz para aplicar la transformación geométrica correspondiente, incluida la traslación cuando la forma de la matriz lo permite.

El argumento de transformación también puede especificarse como una lista con la forma siguiente:

```
{rotation_matrix or -1 [, "N"] [, {eye_x, eye_y, eye_z} or -1] [, {xmin3D, xmax3D, ymin3D, ymax3D, zmin3D, zmax3D}]}
```

- rotation_matrix or -1: Matriz utilizada para transformar los puntos. Especifique -1 cuando no deba aplicarse ninguna matriz de transformación.
- "N": Indicador opcional de normalización. Cuando se especifica, las coordenadas z transformadas se normalizan al intervalo de 0 a 255, lo que permite utilizarlas para el recorte de profundidad.
- {eye_x, eye_y, eye_z} or -1: Posición opcional del observador utilizada para la proyección en perspectiva. Especifique -1 cuando no sea necesaria la proyección en perspectiva.
- {xmin3D, xmax3D, ymin3D, ymax3D, zmin3D, zmax3D}: Límites opcionales que definen la región de recorte 3D.

Después de la transformación, la proyección en perspectiva se aplica mediante las ecuaciones siguientes:

$$x = \text{eye_z}/z * x - \text{eye_x}$$

$$y = \text{eye_z}/z * y - \text{eye_y}$$

Si se proporcionan límites de recorte 3D, cada triángulo se recorta con respecto a la región especificada antes de renderizarse.

Reutilización de puntos transformados

Cuando se aplica una transformación, TRIANGLE devuelve una cadena que contiene los puntos transformados. Esta cadena puede pasarse posteriormente como pre_rotated_points, lo que permite volver a renderizar la misma geometría transformada sin repetir la transformación.

- pre_rotated_points: Cadena de puntos transformados devuelta por una llamada anterior a TRIANGLE.
- triangle_definitions: Definiciones de triángulo que hacen referencia a los puntos almacenados en pre_rotated_points.

La cadena de puntos transformados devuelta también puede reutilizarse mediante LINE cuando se necesiten los mismos puntos y la misma transformación en varias operaciones de dibujo consecutivas.

Recorte de profundidad

TRIANGLE([G]) crea y devuelve una cadena de búfer de profundidad adaptada a las dimensiones de G. A cada píxel del búfer se le asigna inicialmente un valor de profundidad de 255.

Pase esta cadena como zstring junto con valores de profundidad de los vértices comprendidos entre 0 y 255. Durante el renderizado, no se dibujan los píxeles que estén más alejados que los píxeles ya registrados en el búfer. Los píxeles que superan la prueba de profundidad se dibujan y los valores correspondientes de zstring se actualizan automáticamente.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

1281 Dibujar triángulo

ID 1281 · título recuperado de la página española

Sintaxis

```
TRIANGLE_P([G,] x1, y1, x2, y2, x3, y3 [, c [, alpha]])
TRIANGLE_P([G,] x1, y1, x2, y2, x3, y3, c1 [, c2, c3], alpha)
TRIANGLE_P([G,] x1, y1, x2, y2, x3, y3, c1, [c2, c3 [, alpha [, zstring, z1, z2, z3]]])
TRIANGLE_P([G,] {x1, y1 [, c1 [, z1]]}, {x2, y2 [, c2 [, z2]]}, {x3, y3 [, c3 [, z3]]} [, zstring])
TRIANGLE_P([G,] points_definition, triangle_definitions, rotation_matrix or {rotation_matrix or -1, ["N"]}, [{eye_x, eye_y, eye_z} or -1], [{xmin3D, xmax3D, ymin3D, ymax3D, zmin3D, zmax3D}], [zstring])
TRIANGLE_P([G,] pre_rotated_points, triangle_definitions [, zstring])
TRIANGLE_P([G])
```

Dibuja uno o varios triángulos rellenos en el GROB G mediante coordenadas de píxel. Un triángulo puede dibujarse con un único color o con colores independientes para sus vértices. Cuando se especifican tres colores de vértice, estos se interpolan sobre la superficie del triángulo. La transparencia se controla exclusivamente mediante el parámetro independiente alpha; el componente alfa de los valores de color se ignora.

Las formas avanzadas permiten renderizar varios triángulos en una sola operación y aplicar opcionalmente transformaciones geométricas, proyección en perspectiva, recorte 3D, normalización de profundidad y recorte de profundidad por píxel.

Parámetros

- G: GROB de destino.
- x1, y1: coordenadas de píxel del primer vértice.
- x2, y2: coordenadas de píxel del segundo vértice.
- x3, y3: coordenadas de píxel del tercer vértice.
- c: Color opcional aplicado a todo el triángulo.
- c1, c2, c3: Colores asignados a los vértices correspondientes. Si se especifican los tres colores, estos se interpolan sobre la superficie del triángulo.
- alpha: Transparencia opcional aplicada a todo el triángulo, de 0 (completamente opaco) a 255 (completamente transparente).
- z1, z2, z3: Valores de profundidad asignados a los vértices correspondientes. Cuando se utilizan con zstring, estos valores deben estar comprendidos entre 0 y 255.
- zstring: Cadena opcional de búfer de profundidad utilizada para el recorte de profundidad por píxel. Debe crearse mediante TRIANGLE_P([G]) y se actualiza a medida que se dibujan los triángulos.

Renderizado de varios triángulos

- points_definition: Lista o matriz que contiene los puntos utilizados por los triángulos. Cada definición de punto contiene las coordenadas x e y y puede incluir además una coordenada z, un color o ambos. Las coordenadas también pueden proporcionarse como un único valor de punto.
- triangle_definitions: Lista o matriz que describe los triángulos que se van a renderizar. Cada definición de triángulo contiene entre tres y cinco valores con la forma p1, p2, p3 [, c [, alpha]].
- p1, p2, p3: Índices de las tres entradas de points_definition que definen el triángulo.
- c: Color opcional que sustituye los colores almacenados en las definiciones de punto correspondientes. Para especificar alpha sin sustituir los colores de los puntos, utilice -1 para c.

Varias definiciones de triángulo pueden compartir el mismo color y transparencia mediante la forma agrupada {c [, alpha], triangle_1, ..., triangle_n}.

Transformación y proyección

- rotation_matrix: Matriz de transformación con dimensiones comprendidas entre 2×2 y 3×4. Cada punto se multiplica por esta matriz para aplicar la transformación geométrica correspondiente, incluida la traslación cuando la forma de la matriz lo permite.

El argumento de transformación también puede especificarse como una lista con la forma siguiente:

{rotation_matrix or -1 [, "N"] [, {eye_x, eye_y, eye_z} or -1] [, {xmin3D, xmax3D, ymin3D, ymax3D, zmin3D, zmax3D}]}

- rotation_matrix or -1: Matriz utilizada para transformar los puntos. Especifique -1 cuando no deba aplicarse ninguna matriz de transformación.
- "N": Indicador opcional de normalización. Cuando se especifica, las coordenadas z transformadas se normalizan al intervalo de 0 a 255, lo que permite utilizarlas para el recorte de profundidad.
- {eye_x, eye_y, eye_z} or -1: Posición opcional del observador utilizada para la proyección en perspectiva. Especifique -1 cuando no sea necesaria la proyección en perspectiva.
- {xmin3D, xmax3D, ymin3D, ymax3D, zmin3D, zmax3D}: Límites opcionales que definen la región de recorte 3D.

Después de la transformación, la proyección en perspectiva se aplica mediante las ecuaciones siguientes:

$$x = \text{eye_z}/z * x - \text{eye_x}$$
$$y = \text{eye_z}/z * y - \text{eye_y}$$

Si se proporcionan límites de recorte 3D, cada triángulo se recorta con respecto a la región especificada antes de renderizarse.

Reutilización de puntos transformados

Cuando se aplica una transformación, TRIANGLE_P devuelve una cadena que contiene los puntos transformados. Esta cadena puede pasarse posteriormente como pre_rotated_points, lo que permite volver a renderizar la misma geometría transformada sin repetir la transformación.

- pre_rotated_points: Cadena de puntos transformados devuelta por una llamada anterior a TRIANGLE_P.
- triangle_definitions: Definiciones de triángulo que hacen referencia a los puntos almacenados en pre_rotated_points.

La cadena de puntos transformados devuelta también puede reutilizarse mediante LINE_P cuando se necesiten los mismos puntos y la misma transformación en varias operaciones de dibujo consecutivas.

Recorte de profundidad

TRIANGLE_P([G]) crea y devuelve una cadena de búfer de profundidad adaptada a las dimensiones de G. A cada píxel del búfer se le asigna inicialmente un valor de profundidad de 255.

Pase esta cadena como zstring junto con valores de profundidad de los vértices comprendidos entre 0 y 255. Durante el renderizado, no se dibujan los píxeles que estén más alejados que los píxeles ya registrados en el búfer. Los píxeles que superan la prueba de profundidad se dibujan y los valores correspondientes de zstring se actualizan automáticamente.

Para obtener información general sobre el sistema gráfico, incluidos los valores predeterminados de los parámetros y los formatos de entrada admitidos, consulte la sección Comandos de dibujo.

0616 Transposición

ID 616 · título recuperado de la página española

Sintaxis

TRN(matriz)

Matriz de transposición. Si el modo complejo está activado y la matriz contiene elementos complejos, TRN busca la matriz transpuesta conjugada.

0534 Truncar

ID 534 · título recuperado de la página española

Sintaxis

TRUNCATE(value [, places])

TRUNCATE trunca un número sin redondearlo. El número a truncar se especifica mediante value. Si se proporciona el argumento opcional places, los valores positivos o cero truncan value a esa cantidad de decimales, mientras que los valores negativos truncan value al número especificado de dígitos significativos. Si se omite places, TRUNCATE utiliza el número de decimales especificado por la configuración de visualización del sistema actual.

1304 Tipo de objeto

ID 1304 · título recuperado de la página española

Sintaxis

TYPE(object [, mode])

Devuelve el tipo del objeto especificado.

El argumento opcional mode controla el nivel de información de tipo devuelto:

- 0: Devuelve el tipo general del objeto (valor predeterminado si se omite).
- 1: Devuelve información extendida de subtipo, cuando está disponible.

Cuando mode se omite o se establece en 0, TYPE devuelve uno de los siguientes identificadores generales de tipo:

- 0: Número real
- 1: Entero
- 2: Cadena de caracteres
- 3: Número complejo
- 4: Matriz
- 5: Error
- 6: Lista
- 8: Función
- 9: Unidad
- 14.x: Objeto CAS, donde la parte decimal identifica el subtipo CAS

Cuando mode se establece en 1, TYPE intenta devolver información de subtipo más detallada para los tipos de objeto admitidos.

Actualmente, este modo está definido para los números reales y pueden devolverse los siguientes valores de subtipo:

- 0.1: Valor real almacenado en formato BCD
- 0.2: Valor racional exacto almacenado internamente como una fracción, con el numerador y el denominador almacenados como enteros de 64 bits
- 0.3: Múltiplo racional exacto de π , es decir, un valor de la forma $q\pi$, donde q es racional

Para los tipos de objeto para los que no se dispone de información extendida de subtipo, el valor devuelto es el mismo que el devuelto en el modo de tipo general.

0825 Funciones en la vista de Inicio (PPL): 'U'

ID 825 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'U'

1310 Conversión de factores de unidades

ID 1310 · título recuperado de la página española

Sintaxis

UFACTOR(Valor_unidad1, 1_Unidad2)

Convierte una medida utilizando una unidad compuesta en una medida expresada en unidades constituyentes.

Ejemplo: un culombio (medida de carga eléctrica) es una unidad constituyente derivada del amperio, unidad básica del sistema internacional, y en segundo lugar: $1\text{ C} = 1\text{ A} \cdot 1\text{ s}$. Mediante UFACTOR, puede expresar una medida en culombios como un producto de amperios y tiempo.

1739 UPART — Devuelve la parte de unidad de un objeto de unidad.

ID 1739 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: UPART

Explicación española: Devuelve la parte de unidad de un objeto de unidad.

Sintaxis

UPART(unidad_número)

Devuelve la parte de unidad de un objeto de unidad.

1312 Mayúsculas

ID 1312 · título recuperado de la página española

Sintaxis

UPPER(Cadena)
UPPER(Lista)

Para una cadena, devuelve la cadena con los caracteres en minúscula convertidos a mayúsculas.

Para una lista de cadenas, devuelve la lista de cadenas con los caracteres en minúscula convertidos a mayúsculas.

1727 USBOpen — () cierra cualquier conexión activa y devuelve la lista de los dispositivos HID actualmente conectados.

ID 1727 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: USBOpen

Explicación española: () cierra cualquier conexión activa y devuelve la lista de los dispositivos HID actualmente conectados.

Sintaxis

USBOpen()
USBOpen(vid, pid)

USBOpen() cierra cualquier conexión activa y devuelve la lista de los dispositivos HID actualmente conectados.

USBOpen(vid, pid) Si un dispositivo USB con el vid / pid especificado está conectado a la calculadora, se abre para el envío y recepción de datos a través de las funciones USBSend y USBReceive.

Si la función falla, devuelve 0. De lo contrario, devuelve una lista que contiene las longitudes de los informes de salida e entrada.

1729 USBRecv — Devuelve una lista de números que representa los datos que provienen del dispositivo USB.

ID 1729 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: USBRecv

Explicación española: Devuelve una lista de números que representa los datos que provienen del dispositivo USB.

Sintaxis

USBRecv

Devuelve una lista de números que representa los datos que provienen del dispositivo USB. Si la función falla, cierra el puerto USB y devuelve 0. Si no hay datos entrantes, devolverá una lista vacía. Hay un búfer limitado para los datos entrantes. Por lo tanto, es fundamental llamar a USBRecv con suficiente frecuencia para asegurarse de que no se pierdan datos.

1728 USBSend — Envía un paquete USB compuesto por los números proporcionados en la lista de datos (un elemento es un...

ID 1728 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: USBSend

Explicación española: Envía un paquete USB compuesto por los números proporcionados en la lista de datos (un elemento es un...

Sintaxis

USBSend({datos})

Envía un paquete USB compuesto por los números proporcionados en la lista de datos (un elemento es un byte).

Si los datos son más pequeños que un paquete, el paquete se rellenará con 0.

Si los datos son más grandes que un paquete, solo se enviarán los primeros bytes.

En caso de error, cerrará automáticamente el puerto USB y devolverá 0.

1311 Simplificación de unidades

ID 1311 · título recuperado de la página española

Sintaxis

USIMPLIFY(Valor_Unitsexpr)

Simplifica el valor de una expresión de unidades compleja Unitsexpr en un valor equivalente de una expresión de unidades más sencilla.

Ejemplos

(un julio se define como 1 kg*m²/s²) [Ejemplo gráfico E1] [Ejemplo gráfico E2]

0614 Lista de Union

ID 614 · título recuperado de la página española

Sintaxis

UNION(Object1 [, Object2, ...])

Toma un conjunto de objetos de entrada y los concatena en una lista con duplicados eliminados. Si un objeto de entrada es otra lista, se aplanan en sus elementos constitutivos antes de eliminar duplicados. Para evitar que una lista se aplaste, debe estar anidada (una lista dentro de otra lista, ver ejemplos).

1507 UNTIL — Especifica la condición de terminación para el bucle REPEAT - UNTIL.

ID 1507 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: UNTIL

Explicación española: Especifica la condición de terminación para el bucle REPEAT - UNTIL.

Sintaxis

UNTIL test;

Especifica la condición de terminación para el bucle REPEAT - UNTIL. Consulte la descripción de REPEAT para más detalles.

1738 UVAL — Devuelve la parte numérica del objeto de unidad.

ID 1738 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: UVAL

Explicación española: Devuelve la parte numérica del objeto de unidad.

Sintaxis

UVAL(Unidad_Valor)

Devuelve la parte numérica del objeto de unidad.

0826 Funciones en la vista de Inicio (PPL): 'V'

ID 826 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'V'

1521 VERSION — Devuelve una secuencia que contiene los números de la versión referente a los diversos componentes del...

ID 1521 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: VERSION

Explicación española: Devuelve una secuencia que contiene los números de la versión referente a los diversos componentes del...

Sintaxis

VERSION([n])

Devuelve una secuencia que contiene los números de la versión referente a los diversos componentes del sistema. Esto equivale a la página de ayuda Acerca de Prime

Si hay un entero dado n, devuelve esa parte específica de la secuencia de la versión.

1497 Palabra clave View

ID 1497 · título recuperado de la página española

Sintaxis

```
VIEW "Texto" Función()  
BEGIN  
END;
```

Solo puede utilizarse en un programa de aplicación.

Permite a un programador definir una entrada en el menú «Vistas». El texto «Texto» se muestra junto con las demás vistas definidas cuando se pulsa la tecla View. A continuación, cuando se toca la tecla de menú [[OK]] o se pulsa la tecla Enter, se ejecuta Función para la entrada seleccionada.

Tenga en cuenta que una función VIEW también puede exportarse.

0827 Funciones en la vista de Inicio (PPL): 'W'

ID 827 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'W'

1328 Esperar

ID 1328 · título recuperado de la página española

Sintaxis

WAIT(n)

Detiene la ejecución del programa.

El comportamiento depende del valor de n:

- Si $n \geq 1$: Detiene la ejecución durante n segundos. Se aceptan valores fraccionarios, lo que permite especificar retardos con una precisión cercana a un milisegundo. Devuelve el valor de n.
- Si $n = 0$ o se omite n: Detiene la ejecución hasta que se pulsa una tecla. Si se pulsa una tecla, devuelve su código. Si no se pulsa ninguna tecla en 1 minuto, devuelve -1.
- Si $n = -1$: Detiene la ejecución hasta que se pulsa una tecla, se produce un evento del ratón o transcurre el tiempo de espera de 1 minuto. Si se pulsa una tecla, devuelve su código. Si se produce un evento del ratón, devuelve una lista en una de las siguientes formas: {type} {type, x, y} {type, x, y, dx, dy}

Salvo que se indique lo contrario, x e y especifican la posición del evento.

Si no se pulsa ninguna tecla ni se produce ningún evento del ratón en 1 minuto, devuelve -1.

Para los eventos del ratón, type puede tener uno de los siguientes valores:

0: Botón del ratón pulsado

1: Movimiento del ratón

2: Botón del ratón liberado; no se proporcionan x ni y

3: Clic del ratón; cuando se detecta un clic, no se devuelve un evento independiente de liberación del botón

5: Arrastre del ratón; x e y indican el movimiento desde el evento anterior, mientras que dx y dy indican el movimiento total desde que se pulsó originalmente el botón del ratón

6: Rotación del ratón; x es el ángulo original e y es el nuevo ángulo, expresado en treintaidosavos de una vuelta completa

7: Clic prolongado del ratón; indica que el botón del ratón permaneció pulsado durante 1 segundo

1329 Estructura de bucle while

ID 1329 · título recuperado de la página española

Sintaxis

```
WHILE prueba DO comandos END;
```

Ejecuta comandos WHILE prueba es verdadero.

Código de ejemplo:

```
A:=5;  
WHILE A>0 DO  
PRINT(A);  
A:=A-1;  
END;
```

introduce 5 4 3 2 1

0828 Funciones en la vista de Inicio (PPL): 'X'

ID 828 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con 'X'

1322 XOR lógico

ID 1322 · título recuperado de la página española

Sintaxis

```
Valor1 XOR Valor2
```

Para números reales, devuelve 1 si Valor1 o Valor2 no es un cero, pero no ambos; de lo contrario, devuelve 0. Para enteros y cadenas, XOR se realiza a nivel de bit, y devuelve 1 si, exactamente, un bit es 1 y el bit correspondiente es 0, de lo contrario, devuelve 0.

0536 Exponente

ID 536 · título recuperado de la página española

Sintaxis

```
XPON(valor)
```

Exponente. Devuelve al exponente de valor.

0829 Funciones en la vista de Inicio: símbolos

ID 829 · título recuperado de la página española

Lista de funciones y comandos en la vista de Inicio (PPL) que comienzan con símbolos

0542 Porcentaje

ID 542 · título recuperado de la página española

Sintaxis

```
%(x, y)
```

x por ciento de y.

Devuelve $(x/100)*y$.

1343 Porcentaje de cambio

ID 1343 · título recuperado de la página española

Sintaxis`%CHANGE(x, y)`

Porcentaje de cambio x a y. Devuelve $100 \cdot (y-x)/x$.

1344 Porcentaje Total

ID 1344 · título recuperado de la página española

Sintaxis`%TOTAL(x, y)`

Porcentaje total; el porcentaje de x que es y. Devuelve $100 \cdot y/x$.

0609 Lista Δ

ID 609 · título recuperado de la página española

Sintaxis `$\Delta$ LIST(list)`

Crea una nueva lista compuesta por las primeras diferencias de los elementos de la lista. Cada elemento del resultado se calcula restando el elemento precedente del elemento actual. Por lo tanto, la nueva lista tiene un elemento menos que la lista de entrada.

Los elementos adyacentes de la lista se procesan de izquierda a derecha usando el operador -. Cada par adyacente debe ser compatible con el operador -. Los elementos pueden ser valores simples o listas, incluyendo listas anidadas.

Los valores simples y las listas se combinan según las reglas habituales de la aritmética de listas. Cuando se restan listas entre sí, deben tener tamaños compatibles en cada nivel de anidamiento correspondiente. Los tipos de elementos o estructuras de lista incompatibles producen un error.

El tipo exacto del resultado depende de los tipos de los elementos adyacentes y de cómo se define el operador - para ellos.

0611 Lista Π

ID 611 · título recuperado de la página española

Sintaxis `$\Pi$ LIST(list)`

Calcula el producto de los elementos de la lista de entrada.

Los elementos de list se multiplican de izquierda a derecha utilizando el operador *. Los elementos pueden ser valores individuales (por ejemplo, un número, un vector o una matriz) o listas, incluidas listas anidadas.

Los valores individuales y las listas se combinan según las reglas habituales de la aritmética de listas. Cuando se multiplican listas entre sí, deben tener tamaños compatibles en cada nivel de anidamiento correspondiente. Las estructuras de listas incompatibles producen un error.

0610 Lista Σ

ID 610 · título recuperado de la página española

Sintaxis `$\Sigma$ LIST(list)`

Calcula el resultado de sumar todos los elementos de la lista de entrada.

Los elementos de la lista se suman de izquierda a derecha utilizando el operador +. Cada elemento puede ser un valor simple (por ejemplo, un número, una cadena o una matriz) o una lista, incluyendo una lista anidada.

Los valores simples y las listas se combinan de acuerdo con las reglas aritméticas de listas habituales. Cuando se suman listas, deben tener tamaños compatibles en cada nivel de anidamiento correspondiente. Las estructuras de listas incompatibles producen un error.

El tipo exacto del resultado depende de los tipos de los elementos y de cómo está definido el operador + para ellos.

1422 Agregar.

ID 1422 · título recuperado de la página española

Sintaxis

`Objeto1 + Objeto2`

Devuelve el resultado de sumar Objeto2 a Objeto1. Los objetos pueden ser valores numéricos o expresiones que devuelven resultados numéricos. Los objetos también pueden ser listas o matrices de dimensiones apropiadas.

1421 Restar

ID 1421 · título recuperado de la página española

Sintaxis

`Objeto1 - Objeto2`

Devuelve el resultado de restar Objeto2 de Objeto1. Los objetos pueden ser valores numéricos o expresiones que devuelven resultados numéricos. Los objetos también pueden ser listas o matrices de dimensiones apropiadas.

1417 Dividir

ID 1417 · título recuperado de la página española

Sintaxis

`Objeto1 ÷ Objeto2`

Devuelve el resultado de dividir Objeto1 por Objeto2.

Los objetos pueden ser valores numéricos o expresiones que devuelven resultados numéricos. Los objetos también pueden ser listas o matrices de dimensiones adecuadas.

1420 Multiplicar

ID 1420 · título recuperado de la página española

Sintaxis

`Objeto1×Objeto2`

Devuelve el resultado de multiplicar Objeto1 y Objeto2.

Los objetos pueden ser valores numéricos o expresiones que devuelven resultados numéricos. Los objetos también pueden ser listas o matrices de dimensiones adecuadas.

1433 Inverso

ID 1433 · título recuperado de la página española

Sintaxis

`Objeto□`

Devuelve el inverso de un Objeto. El objeto puede ser un número, una expresión que resulta en un valor numérico, una lista o una matriz cuadrada.

0568 Factorial

ID 568 · título recuperado de la página española

Sintaxis

`valor!`

Para números enteros, calcula el Factorial de valor.

Para números enteros negativos, reales o complejos, calcula la función Gamma: $x! = \Gamma(x + 1)$.

1463 Mayor o igual que

ID 1463 · título recuperado de la página española

Sintaxis

Valor1 ≥ Valor2
Valor1 >= Valor2

Prueba si el Valor1 es o no es mayor o igual que el Valor2. Devuelve 1 si es verdadero o 0 si es falso.

1464 Equivalencia

ID 1464 · título recuperado de la página española

Sintaxis

Valor1 == Valor2

Prueba si Valor1 es igual a Valor2. Devuelve 1 si es verdadero, 0 si es falso. Si se enumeran Valor1 y Valor2, devuelve una lista que contiene 0 o 1 para cada par de elementos.

Si se prueban dos expresiones para ver la equivalencia, los resultados de la prueba son sensibles a la Configuración CAS.

Si el ajuste Simplificar se establece como Ninguno, $a=b$ comprueba que a y b tienen la misma representación de árbol.

Si Simplificar está definido como Máximo, se llama a la función de simplificación automática en $a-b$ y el sistema devuelve verdadero si el resultado de la función Simplificar en $a-b$ es 0.

Nota: cuando la calculadora evalúa una expresión que contiene una prueba, cada prueba se evalúa a medida que se encuentra antes de procesar el resto de la expresión. Por lo tanto, una expresión compuesta como $A==B==C$ evaluará primero $A==B$ y luego comparará el resultado de esa evaluación (0 o 1) con C . En otras palabras, hay paréntesis implícitos alrededor de cada argumento sucesivo, por lo que $A==B==C==D$ se evalúa como $((A==B)==C)==D$.

1462 Mayor que

ID 1462 · título recuperado de la página española

Sintaxis

Valor1 > Valor2

Prueba si el Valor1 es mayor que el Valor2 o no. Devuelve 1 si es verdadero o 0 si es falso.

1460 Menor que

ID 1460 · título recuperado de la página española

Sintaxis

Valor1 < Valor2

Comprueba si Valor1 es menor que Valor2. Devuelve 1 si es verdadero, 0 si es falso.

1461 Menor o igual que

ID 1461 · título recuperado de la página española

Sintaxis

Valor1 ≤ Valor2
Valor1 <= Valor2

Prueba si Valor1 es o no menor o igual a Valor2. Devuelve 1 si es verdadero, 0 si es falso.

1465 No es igual que

ID 1465 · título recuperado de la página española

Sintaxis

Valor1 ≠ Valor2
Valor1 <> Valor2

Prueba si el Valor1 no es igual al Valor2. Devuelve 1 si es verdadero o 0 si es falso.

Nota: cuando la calculadora evalúa una expresión que contiene una prueba, cada prueba se evalúa a medida que se encuentra antes de procesar el resto de la expresión. Por lo tanto, una expresión compuesta como $A \neq B \neq C$ evaluará primero $A \neq B$ y luego comparará el resultado de esa evaluación (0 o 1) con C. En otras palabras, hay paréntesis implícitos alrededor de cada argumento sucesivo, por lo que $A \neq B \neq C \neq D$ se evalúa como $((A \neq B) \neq C) \neq D$.

1434 Ángulo

ID 1434 · título recuperado de la página española

Sintaxis

Valor1 $\angle$ Valor2

$\angle$: Ángulo de un número complejo en modo polar.

Ingresa el símbolo de ángulo. Se utiliza para introducir números complejos en forma polar. Con un número complejo en Ans y en la pantalla como el último resultado, presione esta tecla para alternar entre las formas rectangulares y polares del número complejo.

1016 Asignar

ID 1016 · título recuperado de la página española

Sintaxis

variable := objeto

Asigna objeto a la variable.

Ejemplos

A := 3 almacena el valor 3 en la variable A
F1 := 3-X forma F1(X)=3-X
M5 := [1, 2] almacena un vector en M5

1306 Almacenar

ID 1306 · título recuperado de la página española

Sintaxis

valor ► variable

Almacena el valor en la variable.

Ejemplo

3►A almacena el valor 3 en la variable A.

1337 .* — Realiza una multiplicación elemento por elemento de 2 listas o 2 matrices.

ID 1337 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: .*

Explicación española: Realiza una multiplicación elemento por elemento de 2 listas o 2 matrices.

Sintaxis

Lista1 .* Lista2
Matriz1 .* Matriz2

Realiza una multiplicación elemento por elemento de 2 listas o 2 matrices.

1341 .+ — Agrega un valor (real o complejo) a cada elemento de una lista o matriz.

ID 1341 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: .+

Explicación española: Agrega un valor (real o complejo) a cada elemento de una lista o matriz.

Sintaxis

Matriz .+ Valor
Valor .+ Matriz
Lista .+ Valor
Valor .+ Lista

Agrega un valor (real o complejo) a cada elemento de una lista o matriz.

Tenga en cuenta que cuando el valor precede al operador, debe poner un espacio entre el valor y el operador; de lo contrario, el punto en el operador puede ser malinterpretado como un (potencialmente superfluo) punto decimal en el valor.

1342 .- — Resta un valor (real o complejo) de cada elemento de una lista o vector.

ID 1342 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: .-

Explicación española: Resta un valor (real o complejo) de cada elemento de una lista o vector.

Sintaxis

Matriz .- Valor
Valor .- Matriz
Lista .- Valor
Valor .- Lista

Resta un valor (real o complejo) de cada elemento de una lista o vector. Cuando el valor precede al operador, la operación resta cada elemento de la lista o matriz del valor. En este último caso, debe colocar un espacio entre el valor y el operador; de lo contrario, el punto en el operador puede ser malinterpretado como un (potencialmente superfluo) punto decimal en el valor.

1339 ./ — División de elemento por elemento.

ID 1339 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ./

Explicación española: División de elemento por elemento.

Sintaxis

Lista1 ./ Lista2
Matriz1 ./ Matriz2

División de elemento por elemento. Devuelve la división del término por término de dos listas o dos matrices.

1338 .^ — Devuelve el resultado de elevar Objeto1 a la Objeto2 potencia.

ID 1338 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: .^

Explicación española: Devuelve el resultado de elevar Objeto1 a la Objeto2 potencia.

Sintaxis

Objeto1 .^ Objeto2

Devuelve el resultado de elevar Objeto1 a la Objeto2 potencia.

Uno o ambos objetos son típicamente listas o matrices. Los objetos pueden ser valores numéricos o expresiones que devuelven resultados numéricos.

1436 Tecla igual

Introduce el signo igual (=) en la posición actual del cursor.

El signo igual puede utilizarse de distintas maneras según el contexto actual, por ejemplo, en ecuaciones y expresiones en la Vista de Inicio o la Vista CAS, en definiciones en la Vista simbólica de una aplicación o en programas.

Otros usos del signo igual:

- El operador booleano (lógico) de equivalencia es ==
- El operador mayor o igual que puede introducirse como >=
- El operador menor o igual que puede introducirse como <=
- El operador de asignación es :=

1333 Función Donde

Sintaxis

Expr| Var=Valor

Expr| {Var1=Val1, Var2=Val2, ...}

Sustituye el Valor por la Var en una Expr y evalúa el resultado. Este comando también puede tomar una lista de sustituciones de múltiples variables.

1334 Constante matemática π

La constante π es la relación entre la circunferencia y el diámetro de cualquier círculo. Internamente representada como 3.14159265359.

1335 Símbolo matemático ∞

El símbolo ∞ representa el concepto de una cantidad ilimitada. Cuando se usa con un signo positivo ($+\infty$ o simplemente ∞), denota un valor mayor que cualquier número finito, mientras que un signo negativo ($-\infty$) indica un valor menor que cualquier número finito. El símbolo $\pm\infty$ indica que el valor puede extenderse a infinito positivo o negativo, dependiendo del contexto. Se usa comúnmente en contextos matemáticos para significar límites o valores que se extienden indefinidamente. Representado internamente como un símbolo especial que indica magnitud infinita.

1336 Σ - Summation

Sintaxis

$\Sigma(\text{expr}, \text{var}, \text{valori}, \text{valorf})$

Calcula la suma de expr con respecto a var a medida que var pasa de valori a valorf en incrementos de 1.

1330 ∂ - Derivada numérica

Sintaxis

$\partial(\text{Expr}, \text{Var=Value})$

Devuelve el valor de la derivada de Expr con respecto a Var, evaluada en Value.

El argumento Var=Value especifica tanto la variable de diferenciación como el valor en el que se evalúa la derivada.

En la Vista de Inicio, el resultado se devuelve como una aproximación decimal. En la Vista de CAS, el resultado puede devolverse como un valor exacto.

Este comando puede utilizarse en la Vista simbólica de la aplicación Función para definir una función como una derivada. Por ejemplo, $F1(X)=\partial(1/X-X, X=X)$ traza la gráfica de la derivada de $f(X)=1/X-X$.

Ejemplos

[Ejemplo gráfico E1] devuelve -1,04 en la Vista de Inicio y -26/25 en la Vista de CAS.

1331 $\int$ - Integral numérica

ID 1331 · título recuperado de la página española

Sintaxis

$\int(\text{Expr}, \text{Var}, \text{Val1}, \text{Val2})$

Devuelve la integral de una expresión.

Con una expresión como argumento, devuelve la integral indefinida con respecto a x. Con el segundo, terceros y cuarto argumento opcional, puede especificar la variable de integración y los límites de un integral definida.

1340 Valor imaginario

ID 1340 · título recuperado de la página española

Sintaxis

$\square$

La raíz cuadrada de -1.

0830 Índice de comandos CAS

ID 830 · título localizado en HelpEs

Lista de todas las funciones y comandos disponibles en la vista CAS

0831 Funciones en vista CAS: 'A'

ID 831 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'A'

0883 a2q — Dada una matriz simétrica y un vector de variables, devuelve la forma cuadrática de la matriz usando las...

ID 883 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: a2q

Explicación española: Dada una matriz simétrica y un vector de variables, devuelve la forma cuadrática de la matriz usando las...

Sintaxis

a2q(Matriz, [Var1, Var2, ...])

Dada una matriz simétrica y un vector de variables, devuelve la forma cuadrática de la matriz usando las variables del vector.

0884 abcuv — Dados tres polinomios `A`, `B` y `C`, devuelve U y V tales que $A*U+B*V=C$.

ID 884 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: abcuv

Explicación española: Dados tres polinomios `A`, `B` y `C`, devuelve U y V tales que $A*U+B*V=C$.

Sintaxis

abcuv(A, B, C [, Var])

Dados tres polinomios A, B y C, devuelve U y V tales que $A*U+B*V=C$. Con una variable Var como argumento final, U y V se expresan en términos de esa variable (si es necesario); de lo contrario, se utiliza x.

0885 about — Devuelve la hipótesis hecha con los comandos assume y additionally sobre la variable Var.

ID 885 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: about

Explicación española: Devuelve la hipótesis hecha con los comandos assume y additionally sobre la variable Var.

Sintaxis

about(Var)

Devuelve la hipótesis hecha con los comandos assume y additionally sobre la variable Var.

Ejemplos

[Ejemplo gráfico E1] devuelve las condiciones impuestas en la variable n.

0628 Valor de absoluto

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Valor de absoluto.](#)

0192 Abscisa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Abscisa.](#)

0886 Coseno inverso

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Coseno inverso.](#)

0733 acos2asin — Reemplaza arccos(x) por $\pi/2 - \arcsin(x)$ en el argumento Expr.

ID 733 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: acos2asin

Explicación española: Reemplaza arccos(x) por $\pi/2 - \arcsin(x)$ en el argumento Expr.

Sintaxis

acos2asin(Expr)

Reemplaza arccos(x) por $\pi/2 - \arcsin(x)$ en el argumento Expr.

0734 acos2atan — Sustituye arccos(x) por $\pi/2 - \arctan(x/\sqrt{1-x^2})$ en el argumento.

ID 734 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: acos2atan

Explicación española: Sustituye arccos(x) por $\pi/2 - \arctan(x/\sqrt{1-x^2})$ en el argumento.

Sintaxis

acos2atan(Expr)

Sustituye arccos(x) por $\pi/2 - \arctan(x/\sqrt{1-x^2})$ en el argumento.

0564 Coseno hiperbólico inverso

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Coseno hiperbólico inverso.](#)

0559 Arco cotangente

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Arco cotangente.](#)

0555 Arco cosecante

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Arco cosecante.](#)

0888 additionally — Se usan en la programación con `assume()` para indicar una hipótesis adicional sobre una variable.

ID 888 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: additionally

Explicación española: Se usan en la programación con `assume()` para indicar una hipótesis adicional sobre una variable.

Sintaxis

additionally(Expr)

Se usan en la programación con `asume()` para indicar una hipótesis adicional sobre una variable.

1686 Matriz adjunta

ID 1686 · título recuperado de la página española

Sintaxis

`adjoint_matrix(matriz)`

Devuelve el polinomio característico de A y la comatriz de A-xI.

0193 Affix

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Affix.](#)

0892 algvar — Devuelve una matriz de los nombres de variable simbólica utilizado en una expresión.

ID 892 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `algvar`

Explicación española: Devuelve una matriz de los nombres de variable simbólica utilizado en una expresión.

Sintaxis

`algvar(Expr)`

Devuelve una matriz de los nombres de variable simbólica utilizado en una expresión. La lista se ordena según las extensiones algebraicas requeridas para construir la expresión original.

0893 alog10 — Función $x \rightarrow 10^x$.

ID 893 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `alog10`

Explicación española: Función $x \rightarrow 10^x$.

Sintaxis

`alog10(Expr)`

Función $x \rightarrow 10^x$.

0894 altitude — Dados tres puntos no lineales, dibuja altitude del triángulo definido por los tres puntos que pasan a...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de altitude.](#)

0203 Ángulo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ángulo.](#)

0896 angleat — Utilizado en la Vista simbólica de la aplicación Geometría.

ID 896 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `angleat`

Explicación española: Utilizado en la Vista simbólica de la aplicación Geometría.

Sintaxis

`angleat(punto1, punto2, punto3, punto4)`

Utilizado en la Vista simbólica de la aplicación Geometría.

Dados tres puntos de un ángulo y un cuarto punto como una ubicación, muestra la medida del ángulo definido por los tres primeros puntos. La medida se muestra, con una etiqueta, en la ubicación de la Vista gráfica dada por el cuarto punto. El primer punto es el vértice del ángulo.

Ejemplos

[Ejemplo gráfico E1] (en el modo grado) muestra “appoint(0,0)=40,9” en el punto (-6,6)

0897 Última respuesta

ID 897 · título recuperado de la página española

Sintaxis

Ans

En la vista de Inicio Ans devuelve el resultado del último cálculo efectuado en la vista de Inicio con total precisión. La variable Ans es distinta de los números que aparecen en el historial de Inicio. Los valores de Ans se almacenan internamente con total precisión, mientras que los números visualizados coinciden con el modo de presentación. Ans(n) devuelve la n-ésima entrada en el historial de la vista de Inicio.

En la Vista CAS, Ans devuelve el último resultado en el historial de CAS y Ans(n) no recupera el enésimo elemento del historial. Aquí, Ans(n) tratará de sustituir n por x (o la variable predefinida) en el último elemento del historial y devuelve el resultado. En la Vista CAS, si Ans es una matriz, Ans(m,n) devuelve el elemento en la fila m y la columna n.

0898 Agregar

ID 898 · título recuperado de la página española

Sintaxis

append(Lista, Elemento)
append(Vector, Elemento)

Agregar un elemento a una lista o vector.

0899 apply — Devuelve un vector o matriz que contiene los resultados de aplicar la función f a los elementos en el...

ID 899 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: apply

Explicación española: Devuelve un vector o matriz que contiene los resultados de aplicar la función f a los elementos en el...

Sintaxis

apply(Var→f(Var), Vector)
apply(Var→f(Var), Matriz)

Devuelve un vector o matriz que contiene los resultados de aplicar la función f a los elementos en el vector o la matriz.

0900 approx — Utilizado en el CAS para devolver la evaluación numérica del primer argumento con el número de dígitos...

ID 900 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: approx

Explicación española: Utilizado en el CAS para devolver la evaluación numérica del primer argumento con el número de dígitos...

Sintaxis

approx(Expr [, Ent])

Utilizado en el CAS para devolver la evaluación numérica del primer argumento con el número de dígitos como segundo argumento.

0904 arc — Dibuja un arco de círculo dados 2 vértices y el ángulo en el centro.

ID 904 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: arc

Explicación española: Dibuja un arco de círculo dados 2 vértices y el ángulo en el centro.

Sintaxis

`arc(Pnt, Pnt, Real, [Var(C)], [Var(r)])`

Dibuja un arco de círculo dados 2 vértices y el ángulo en el centro. El centro se almacenará en C y el radio en r.

0205 Arc Length

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Arc Length.](#)

0202 Área

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Área.](#)

0903 areaat — Utilizado en la Vista simbólica de la aplicación Geometría.

ID 903 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: areaat

Explicación española: Utilizado en la Vista simbólica de la aplicación Geometría.

Sintaxis

`areaat(Polígono, Punto)`
`areaat(Círculo, Punto)`

Utilizado en la Vista simbólica de la aplicación Geometría.

Muestra el área algebraica de un círculo o un polígono. La medida se muestra, con una etiqueta, en un punto específico de la Vista gráfica.

Ejemplos

[Ejemplo gráfico E1] muestra “ $\text{circle}(x^2+y^2=1)=\pi$ ” en el punto $(-4,4)$

0544 Argumento

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Argumento.](#)

0905 ASC — Devuelve una lista que contiene los valores Unicode numéricos de la cadena.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ASC.](#)

0557 Arco secante

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Arco secante.](#)

0906 Seno inverso

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Seno inverso.](#)

0729 asin2acos — Reemplaza $\arcsin(x)$ por $\pi/2 - \arccos(x)$ en Expr.

ID 729 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: asin2acos

Explicación española: Reemplaza $\arcsin(x)$ por $\pi/2 - \arccos(x)$ en Expr.

Sintaxis

asin2acos(Expr)

Reemplaza $\arcsin(x)$ por $\pi/2 - \arccos(x)$ en Expr.

0730 asin2atan — Sustituye $\arcsin(x)$ por $\arctan(x/\sqrt{1-x^2})$ en Expr.

ID 730 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: asin2atan

Explicación española: Sustituye $\arcsin(x)$ por $\arctan(x/\sqrt{1-x^2})$ en Expr.

Sintaxis

asin2atan(Expr)

Sustituye $\arcsin(x)$ por $\arctan(x/\sqrt{1-x^2})$ en Expr.

0562 Seno hiperbólico inverso

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Seno hiperbólico inverso.](#)

0907 assume — Realiza una suposición en una variable.

ID 907 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: assume

Explicación española: Realiza una suposición en una variable.

Sintaxis

assume(Expr)

Realiza una suposición en una variable.

Ejemplos

[Ejemplo gráfico E1] . Ahora $\text{solve}(a^2=9,a)$ devolverá $\{3\}$ en lugar de $\{-3,3\}$.

0908 Tangente inversa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tangente inversa.](#)

0566 Tangente hiperbólica inversa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tangente hiperbólica inversa.](#)

0741 atan2 — Devuelve el ángulo principal del punto (x,y), utilizando los signos de ambos argumentos para determinar...

ID 741 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: atan2

Explicación española: Devuelve el ángulo principal del punto (x,y), utilizando los signos de ambos argumentos para determinar...

Sintaxis

atan2(y, x)

Devuelve el ángulo principal del punto (x,y), utilizando los signos de ambos argumentos para determinar el cuadrante correcto. El resultado se expresa en la unidad angular actual.

0738 atan2acos — Sustituye $\arctan(x)$ por $\pi/2 - \arccos(x/\sqrt{1+x^2})$ en el argumento.

ID 738 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: atan2acos

Explicación española: Sustituye $\arctan(x)$ por $\pi/2 - \arccos(x/\sqrt{1+x^2})$ en el argumento.**Sintaxis**

atan2acos(Expr)

Sustituye $\arctan(x)$ por $\pi/2 - \arccos(x/\sqrt{1+x^2})$ en el argumento.

0737 atan2asin — Sustituye $\arctan(x)$ por $\arcsin(x/\sqrt{1+x^2})$ en el argumento Expr.

ID 737 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: atan2asin

Explicación española: Sustituye $\arctan(x)$ por $\arcsin(x/\sqrt{1+x^2})$ en el argumento Expr.**Sintaxis**

atan2asin(Expr)

Sustituye $\arctan(x)$ por $\arcsin(x/\sqrt{1+x^2})$ en el argumento Expr.

0746 atrig2ln — Devuelve una expresión con las funciones de trigonométricas inversas usando la función de logaritmo...

ID 746 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: atrig2ln

Explicación española: Devuelve una expresión con las funciones de trigonométricas inversas usando la función de logaritmo...

Sintaxis

atrig2ln(Expr)

Devuelve una expresión con las funciones de trigonométricas inversas usando la función de logaritmo natural.

0832 Funciones en vista CAS: 'B'

ID 832 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'B'

1006 Función barycenter

ID 1006 · título recuperado de la página española

Sintaxis

barycenter([Punto1, Peso1], [Punto2, Peso2], ..., [PuntoN, Peso3])

Calcula el centro hipotético de masa de un conjunto de puntos, cada uno de ellos con un peso dado (un número real). Cada par de punto-peso se incluye entre corchetes como un vector.

0909 basis — Devuelve una base del espacio vectorial generado por los vectores en ``vectors``.

ID 909 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `basis`

Explicación española: Devuelve una base del espacio vectorial generado por los vectores en ``vectors``.

Sintaxis

`basis(vectors)`

Devuelve una base del espacio vectorial generado por los vectores en `vectors`. Si `vectors` es una matriz, sus filas se tratan como vectores. Si `vectors` es una lista o conjunto de vectores, cada elemento se trata como un vector. Todos los vectores deben tener la misma dimensión.

1684 Número Bernoulli

ID 1684 · título recuperado de la página española

Sintaxis

`bernoulli(n, [variable])`

Devuelve el número de Bernoulli, o polinomio de Bernoulli n utilizando `variable`.

0663 Beta — Devuelve el valor de la función Beta para dos valores, x e y , definido como $\text{Gamma}(x) \cdot \text{Gamma}(y) / \text{Gamma}(x+y)$.

ID 663 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `Beta`

Explicación española: Devuelve el valor de la función Beta para dos valores, x e y , definido como $\text{Gamma}(x) \cdot \text{Gamma}(y) / \text{Gamma}(x+y)$.

Sintaxis

`Beta(x, y)`

Devuelve el valor de la función Beta para dos valores, x e y , definido como $\text{Gamma}(x) \cdot \text{Gamma}(y) / \text{Gamma}(x+y)$.

0911 Densidad de probabilidad beta

ID 911 · título recuperado de la página española

Sintaxis

`betad( $\alpha$ ,  $\beta$ ,  $x$ )`

Función de densidad de probabilidad beta

Calcula la densidad de probabilidad de la distribución beta en x para los parámetros α y β .

0912 Beta acumulada

ID 912 · título recuperado de la página española

Sintaxis

`betad_cdf(a, b, x, [x2])`

Devuelve la probabilidad de cola inferior de la función de densidad de probabilidad beta para el valor x , con los parámetros a y b .

0913 Beta acumulada inversa

ID 913 · título recuperado de la página española

Sintaxis

`betad_icdf(a, b, p)`

Devuelve el valor de x de forma tal que la probabilidad de cola inferior beta de x , con los parámetros a y b , sea p .

0914 Entero a cadena binaria

ID 914 · título recuperado de la página española

Sintaxis

`bin(Integer)`

Convierte un entero en una cadena que representa la forma binaria del argumento `Integer`, precedida por "0b".

0582 Densidad de probabilidad binomial

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Densidad de probabilidad binomial.](#)

0590 Binomial acumulada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Binomial acumulada.](#)

0599 Binomial acumulada inversa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Binomial acumulada inversa.](#)

0148 Bisector

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Bisector.](#)

1687 Bloque de matriz

ID 1687 · título recuperado de la página española

Sintaxis

`blockmatrix(m, n, {bloques_de_matriz})`

Devuelve la matriz de tamaño $m \times n$ creada a partir de un vector de `bloques_de_matriz` con recuento $m \times n$.

1744 bool — Convierte cualquier argumento dado en un valor booleano.

ID 1744 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `bool`

Explicación española: Convierte cualquier argumento dado en un valor booleano.

Sintaxis

`bool(obj)`

Convierte cualquier argumento dado en un valor booleano. Devuelve `true` si el argumento `obj` es distinto de cero o no está vacío. Si `obj` es 0 o un contenedor vacío (como una lista vacía `{}` o un vector vacío `[]`), la función devuelve `false`. Sin embargo, para las cadenas, siempre devuelve `true`, incluso si la cadena está vacía.

0926 breakpoint — Agrega un punto de interrupción.

ID 926 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: breakpoint

Explicación española: Agrega un punto de interrupción.

Sintaxis

breakpoint(Ent)

Agrega un punto de interrupción.

0833 Funciones en vista CAS: 'C'

ID 833 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'C'

1705 c1oc2 — Devuelve el producto de permutación de ciclo1 y ciclo2.

ID 1705 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: c1oc2

Explicación española: Devuelve el producto de permutación de ciclo1 y ciclo2.

Sintaxis

c1oc2(ciclo1, ciclo2)

Devuelve el producto de permutación de ciclo1 y ciclo2.

1704 c1op2 — Devuelve el producto de permutación de ciclo and permutación.

ID 1704 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: c1op2

Explicación española: Devuelve el producto de permutación de ciclo and permutación.

Sintaxis

c1op2(ciclo, permutación)

Devuelve el producto de permutación de ciclo and permutación.

0928 Forma canónica

ID 928 · título recuperado de la página española

Sintaxis

canonical_form(Trinomio, [Var])

Forma canónica de un polinomio de segundo grado.

0931 Concatenar

ID 931 · título recuperado de la página española

Sintaxis

cat(Obj1, Obj2, ..., Objn)

Evalúa los objetos en una secuencia, luego los devuelve concatenados en forma de cadena.

0932 Densidad Cauchy

ID 932 · título recuperado de la página española

Sintaxis`cauchy([x0], [a], x)`

Función de densidad de probabilidad de Cauchy

Calcula la densidad de probabilidad de la distribución de Cauchy en x, con los parámetros x0 y a. De forma predeterminada, x0 es 0 y a es 1.

0933 Cauchy acumulada

ID 933 · título recuperado de la página española

Sintaxis`cauchy_cdf(x0, a, x, [x2])`

Devuelve la probabilidad de cola más baja de la función de densidad de probabilidad de Cauchy para el valor x, dados los parámetros x0 y a.

0934 Cauchy acumulada inversa

ID 934 · título recuperado de la página española

Sintaxis`cauchy_icdf(x0, a, p)`

Devuelve el valor de x de forma que la probabilidad de cola más baja Cauchy de x, dados los parámetros x0 y a, sea p.

0529 Techo**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Techo.](#)

0134 Centro**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Centro.](#)

0935 Factor complejo

ID 935 · título recuperado de la página española

Sintaxis`cFactor(Expr)`

Devuelve una expresión factorizada en el campo complejo (en enteros de Gauss, si hay más de dos).

1688 Matriz equivalente

ID 1688 · título recuperado de la página española

Sintaxis`changebase(A, P)`

changebase toma una matriz cuadrada A y una matriz invertible de cambio de base P, y devuelve la matriz B que representa la misma transformación lineal después del cambio de base definido por P.

El resultado es: $B = \text{inv}(P) * A * P$

1689 Elección de elemento

ID 1689 · título recuperado de la página española

Sintaxis

```
choice(List)  
choice(Vector)
```

Devuelve un elemento aleatorio de una lista o vector no vacío.

0937 Polinomio característico

ID 937 · título recuperado de la página española

Sintaxis

```
charpoly(Matriz, [Var])
```

Devuelve los coeficientes del polinomio característico de una matriz. Con un solo argumento, la variable utilizada en el polinomio es x. Con una variable como segundo argumento, el polinomio devuelto depende de dicha variable.

0788 Resto chino

ID 788 · título recuperado de la página española

Sintaxis

```
chinrem(Matrix1, Matrix2)
```

Calcula el teorema del resto chino para polinomios. Dadas dos matrices de entrada [A,P] y [B,Q], donde P y Q son polinomios coprimos, devuelve [C,P*Q], donde $C = A \bmod P$ y $C = B \bmod Q$.

Cada matriz de entrada debe tener 2 filas: la primera fila contiene los coeficientes del polinomio resto, y la segunda fila contiene los coeficientes del polinomio módulo.

0580 Densidad χ^2 **Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Densidad \$\chi^2\$.](#)

0588 Acumulada χ^2 **Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Acumulada \$\chi^2\$.](#)

0596 Acumulada inversa χ^2 **Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Acumulada inversa \$\chi^2\$.](#)

1654 Prueba de igualdad χ^2

ID 1654 · título recuperado de la página española

Sintaxis

```
chisquaret(Datos, [ley_de_distribución], [parámetros_de_distribución])
```

Prueba de igualdad χ^2 entre 2 (o n) muestras, o entre 1 muestra y ley de distribución.

0643 Cholesky

ID 643 · título recuperado de la página española

Sintaxis

```
cholesky(matriz)
```

Para una matriz simétrica numérica A, devuelve la matriz L de forma que $A=L \cdot \text{tran}(L)$.

0939 Resto chino

ID 939 · título recuperado de la página española

Sintaxis

```
chrem(Lista1, Lista2)
chrem(Vector1, Vector2)
```

Devuelve un vector que contiene los restos chinos para dos conjuntos de números entero, contenidos en dos vectores o dos listas.

0669 Integral del coseno

ID 669 · título recuperado de la página española

Sintaxis

```
Ci(Expr)
```

Devuelve la integral de coseno de una expresión. $\text{int}(\cos(t)/t, t=-\infty..x)$.

0161 Círculo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Círculo.](#)

0162 Circumcircle

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Circumcircle.](#)

0770 Coeficientes del polinomio

ID 770 · título recuperado de la página española

Sintaxis

```
coeff(Expr, [Var], [Entero])
```

Devuelve la lista de coeficientes de un polinomio en relación con el segundo argumento o el coeficiente del término cuyo grado es Entero.

0940 Columna de la matriz

ID 940 · título recuperado de la página española

Sintaxis

```
col(Matriz, Entero)
col(Matriz, Intervalo)
```

Devuelve la columna n o la secuencia de las columnas n1 ... n2 de la matriz A.

0942 Dimensión de la columna

ID 942 · título recuperado de la página española

Sintaxis

```
colDim(Matriz)
```

Devuelve el número de columnas de una matriz.

0675 Recoger como términos

ID 675 · título recuperado de la página española

Sintaxis

```
collect(Poli)
collect(Poli, Var)
collect({Poli1, Poli2, ..., PoliN})
```

Recoge como términos en una expresión polinómica (o de una lista de expresiones polinómicas). Factoriza los resultados, dependiendo de la configuración de CAS.

Si se especifica, recogerá con respecto a Var.

0941 Norma de columna

ID 941 · título recuperado de la página española

Sintaxis

`colnorm(Matriz)`

Encuentra el valor máximo (de todas las columnas) de las sumas de los valores absolutos de todos los elementos en una columna.

1690 Subespacio de la columna

ID 1690 · título recuperado de la página española

Sintaxis

`colspace(matriz [, variable])`

Devuelve una matriz donde las columnas son una base del espacio vectorial generado por las columnas de la matriz A. Si se da, la dimensión de este espacio se almacenará en variable.

0943 Combinaciones

ID 943 · título recuperado de la página española

Sintaxis

`comb(n, r)`

El número de combinaciones de n elementos tomados de r en r, sin tener en cuenta el orden.

Ejemplos

[Ejemplo gráfico E1] (elegir 2 elementos de 5, ignorando el orden)

0944 Denominador común

ID 944 · título recuperado de la página española

Sintaxis

`comDenom(Expr [, Var])`

Reescribe una suma de fracciones racionales como una fracción racional. El denominador de la fracción racional es el común denominador de las fracciones racionales en la expresión original. Con una variable como segundo argumento, el numerador y el denominador se desarrollan en función de esta.

0945 Perpendicular común

ID 945 · título recuperado de la página española

Sintaxis

`common_perpendicular(Línea(D1), Línea(D2))`

Dibuja la perpendicular común de las líneas D1 y D2.

0946 Matriz asociada

ID 946 · título recuperado de la página española

Sintaxis

`companion(Poli, Var)`

Matriz asociada de un polinomio (an=1).

0947 Comparar objetos

ID 947 · título recuperado de la página española

Sintaxis

```
compare(Obj1, Obj2)
```

Compara dos objetos y devuelve 1 si `type(Obj1)<type(Obj2)`, o si `type(Obj1)==type(Obj2)` y `Obj1<Obj2`; de lo contrario, devuelve 0.

1745 complex — Devuelve un número complejo con el valor ``real` + `imag`*i` o convierte una cadena ``str`` o un número en...

ID 1745 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `complex`

Explicación española: Devuelve un número complejo con el valor ``real` + `imag`*i` o convierte una cadena ``str`` o un número en...

Sintaxis

```
complex([real [, imag]])  
complex(str)
```

Devuelve un número complejo con el valor `real + imag*i` o convierte una cadena `str` o un número en un número complejo. Si el primer parámetro es una cadena, se interpretará como un número complejo y la función debe ser llamada sin un segundo parámetro. El segundo parámetro nunca puede ser una cadena. Cada argumento puede ser de cualquier tipo numérico (incluido complejo). Si se omite `imag`, por defecto es cero.

0948 Raíz compleja

ID 948 · título recuperado de la página española

Sintaxis

```
complexroot(Poli, Real, [Complx1, Complx2])
```

Con un polinomio y un real como sus dos argumentos, devuelve una matriz. Cada fila de la matriz contiene una raíz compleja del polinomio con su multiplicidad o un intervalo que contiene dicha raíz y su multiplicidad. El intervalo define una región rectangular (posiblemente) en el plano complejo donde se encuentra una raíz compleja.

Con dos números complejos adicionales como tercer y cuarto argumentos, devuelve una matriz como se describe para dos argumentos, pero solo para las raíces que se extienden en la región rectangular definida por la diagonal creada por los dos números complejos.

0949 Concatenar objetos

ID 949 · título recuperado de la página española

Sintaxis

```
concat(Obj_1, Obj_2 [, Obj_3, ...Obj_N])
```

Toma un conjunto de objetos de entrada y los concatena en una cadena, lista, vector o matriz.

0170 Cónica

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Cónica.](#)

0950 Complejo conjugado

ID 950 · título recuperado de la página española

Sintaxis

```
conj(Compleja)
conj(Lista)
conj(Matriz)
```

La conjugación es la negación (inversión del signo) de la parte imaginaria de un número complejo.

Para una lista o matriz, devuelve una lista o una matriz que contiene los complejos conjugados de todos los elementos complejos.

0951 Contains

ID 951 · título recuperado de la página española

Sintaxis

```
contains(List, Element)
contains(Vector, Element)
contains(String_1, String_2)
```

Dada una lista, un vector o una cadena y un elemento, devuelve el índice de la primera aparición del elemento en la lista, el vector o la cadena. Si el elemento no aparece en la lista, el vector o la cadena, devuelve 0.

0786 Coeficiente de GCD

ID 786 · título recuperado de la página española

Sintaxis

```
content(Poli, [Var])
```

Devuelve el máximo común divisor (MCD) de los coeficientes de un polinomio.

1746 convert — Realiza varios tipos de conversiones dependiendo de los argumentos proporcionados.

ID 1746 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: convert

Explicación española: Realiza varios tipos de conversiones dependiendo de los argumentos proporcionados.

Sintaxis

```
convert(n, base, b)
convert(Value_Unit, _Unit)
convert(r, confrac [, 'Var'])
convert(Expr, Keyword)
```

Realiza varios tipos de conversiones dependiendo de los argumentos proporcionados. Es capaz de manejar conversiones de base numérica, conversiones de unidades, fracciones continuas y transformaciones de múltiples expresiones.

- Conversión de base numérica: Convierte un entero `n` en una lista de sus coeficientes en una base `b`, utilizando 'base' como palabra clave.
- Convierte un valor con una unidad de origen `Value_Unit` a una unidad de destino `_Unit` diferente pero compatible. Por ejemplo, la función puede convertir temperaturas de Celsius a Fahrenheit o longitudes de metros a pies.
- Conversión de fracción continua: Representa un número real `r` como una fracción continua. El segundo argumento debe ser 'confrac'. Opcionalmente, se puede citar un nombre de variable 'Var' para almacenar el resultado.

- **Conversión de expresión:** Convierte una expresión Expr según una palabra clave especificada Keyword: 'string': Convierte una expresión en formato de cadena. 'matrix': Convierte una lista de listas en una matriz. 'sin': Convierte una expresión que involucra el seno trigonométrico (trigsin). 'cos': Convierte una expresión que involucra el coseno trigonométrico (trigcos). 'sincos': Convierte expresiones que involucran tanto el seno como el coseno. 'trig': Convierte una expresión en formato trigonométrico. 'tan': Convierte una expresión que involucra la tangente trigonométrica (half tan). 'exp': Convierte una expresión de formato trigonométrico a formato exponencial (trig2exp). 'ln': Convierte una expresión de formato exponencial a formato de logaritmo natural. 'expln': Convierte expresiones combinadas de formas exponenciales y logarítmicas. 'polynom': Convierte una serie de Taylor en un polinomio eliminando el residuo (order_size). 'partfrac': Convierte una fracción racional en su descomposición en fracciones parciales.

0954 Convex Hull

ID 954 · título recuperado de la página española

Sintaxis

convexhull(Punto1, Punto2, ..., PuntoN)

Devuelve un vector que contiene los puntos que sirven como la envolvente convexa para un conjunto de puntos dado.

0955 Coordenadas

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Coordenadas.](#)

0956 Copiar variable

ID 956 · título recuperado de la página española

Sintaxis

CopyVar(Var1, Var2)

Copia la primera variable en la segunda variable sin evaluación.

0957 Correlación

ID 957 · título recuperado de la página española

Sintaxis

correlation(Lista)
correlation(Matriz)

Devuelve la correlación de los elementos de una lista o matriz.

0958 Función coseno

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función coseno.](#)

0735 cos2sintan — Reemplaza cos(x) por sin(x)/tan(x) en el argumento.

ID 735 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: cos2sintan

Explicación española: Reemplaza cos(x) por sin(x)/tan(x) en el argumento.

Sintaxis

cos2sintan(Expr)

Reemplaza cos(x) por sin(x)/tan(x) en el argumento.

0563 Coseno hiperbólico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Coseno hiperbólico.](#)

0558 Cotangente

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Cotangente.](#)

0959 Contar

ID 959 · título recuperado de la página española

Sintaxis

```
count(Var→Función, Lista)
count(Var→Función, Matriz)
count(Var→Prueba, Lista)
count(Var→Prueba, Matriz)
```

Hay dos usos para esta función, cuyo primer argumento siempre es una asignación de una variable en una expresión.

Si la expresión es una función de la variable, entonces la función se aplica a cada elemento en una lista o una matriz (el segundo argumento) y se devuelve la suma de los resultados.

Si la expresión es una prueba de tipo Booleano, entonces cada elemento en una lista o matriz se prueba y se devuelve el número de elementos que pasan la prueba.

1708 Cuenta elementos iguales

ID 1708 · título recuperado de la página española

Sintaxis

```
count_eq(elemento, lista)
count_eq(elemento, matriz)
```

Devuelve el recuento de elementos de lista o matriz que son iguales a elemento.

Esto es equivalente a contar ($x \rightarrow x == \text{elemento}$, lista o matriz).

1709 Cuenta elementos menores que

ID 1709 · título recuperado de la página española

Sintaxis

```
count_inf(elemento, lista)
count_inf(elemento, matriz)
```

Devuelve el recuento de elementos de lista o matriz que son menores que elemento.

Esto es equivalente a contar ($x \rightarrow x < \text{elemento}$, lista o matriz).

1710 Contar elementos mayores que

ID 1710 · título recuperado de la página española

Sintaxis

```
count_sup(elemento, lista)
count_sup(elemento, matriz)
```

Devuelve el recuento de elementos de lista o matriz que son mayores que elemento.

Esto es equivalente a contar ($x \rightarrow x > \text{elemento}$, lista o matriz).

0960 Covarianza de elementos

ID 960 · título recuperado de la página española

Sintaxis

```
covariance(Lista)
covariance(Matriz)
```

Devuelve la covarianza de los elementos de una lista o una matriz.

0961 Covarianza y correlación

ID 961 · título recuperado de la página española

Sintaxis

```
covariance_correlation(Lista)
covariance_correlation(Matriz)
```

Devuelve un vector que contiene la covarianza y la correlación de los elementos de una lista o vector.

0962 Fracción parcial compleja

ID 962 · título recuperado de la página española

Sintaxis

```
cpartfrac(RatFrac)
```

Devuelve el resultado de la descomposición parcial fracción de una fracción racional en el campo complejo.

0963 Raíces racionales complejas

ID 963 · título recuperado de la página española

Sintaxis

```
crationalroot(Poli)
```

Devuelve la lista de raíces racionales complejas de un polinomio sin indicar la multiplicidad.

0657 Producto vectorial**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Producto vectorial.](#)

0554 Cosecante**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Cosecante.](#)

0712 Solución de complejo

ID 712 · título recuperado de la página española

Sintaxis

```
cSolve(Expr, [Var])
```

Devuelve las soluciones, incluyendo soluciones complejas, de Expr, para Var.

Si Expr es una expresión, resuelve la ecuación Expr=0.

0965 Sumas acumuladas

ID 965 · título recuperado de la página española

Sintaxis

```
cumSum(Lista)
cumSum(Vector)
```

Acepta como argumento una lista o un vector y devuelve una lista o vector cuyos elementos son las sumas acumuladas del argumento original.

0692 Llave rotacional

ID 692 · título recuperado de la página española

Sintaxis

`curl([Expr1, Expr2, ... ExprN], [Var1, Var2, ... VarN])`

Devuelve la llave rotacional de un campo vectorial.

`curl([A,B,C],[x,y,z])` se define como $[dC/dy-dB/dz, dA/dz-dC/dx, dB/dx-dA/dy]$.

0964 Curva

ID 964 · título recuperado de la página española

Sintaxis

`curve(Expr)`

Palabra reservada. No debe utilizarse para nada.

1697 cycle2perm — Convierte ciclo en una permutación.

ID 1697 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `cycle2perm`

Explicación española: Convierte ciclo en una permutación.

Sintaxis

`cycle2perm(ciclo)`

Convierte ciclo en una permutación.

1706 cycleinv — Devuelve el ciclo inverso de ciclo.

ID 1706 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `cycleinv`

Explicación española: Devuelve el ciclo inverso de ciclo.

Sintaxis

`cycleinv(ciclo)`

Devuelve el ciclo inverso de ciclo.

1695 cycles2permu — Convertir un producto de ciclos en una permutación.

ID 1695 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `cycles2permu`

Explicación española: Convertir un producto de ciclos en una permutación.

Sintaxis

`cycles2permu(lista)`

Convertir un producto de ciclos en una permutación.

0790 Ciclotómico

ID 790 · título recuperado de la página española

Sintaxis

`cyclotomic(Entero)`

Genera un vector que representa el enésimo polinomio ciclotómico.

0714 Ceros complejos

ID 714 · título recuperado de la página española

Sintaxis

```
cZeros(Expr [, Var])  
cZeros(Exprs, Vars)
```

El comando `cZeros` devuelve las raíces de la expresión especificada por `Expr`, incluyendo raíces complejas. En esta forma, el resultado es la solución de la ecuación $\text{Expr} = 0$. Si se proporciona el argumento opcional `Var`, especifica la variable a resolver.

El comando `cZeros` también puede resolver un sistema de ecuaciones. En esta forma, `Exprs` especifica una lista o vector de expresiones o ecuaciones, y `Vars` especifica la lista o vector correspondiente de variables a resolver. El resultado es una matriz cuyas filas son las soluciones del sistema.

0834 Funciones en vista CAS: 'D'

ID 834 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'D'

0784 Grado de un polinomio

ID 784 · título recuperado de la página española

Sintaxis

```
degree(Poli)
```

Devuelve el grado de un polinomio.

0969 Eliminar columnas

ID 969 · título recuperado de la página española

Sintaxis

```
delcols(Matriz, Entero)  
delcols(Matriz, Ent1..Ent2)
```

Dada una matriz y un entero n , elimina la n -ésima columna de la matriz y devuelve el resultado. Si se utiliza un intervalo de dos enteros en lugar de un único entero, elimina todas las columnas en el intervalo y devuelve el resultado.

0971 Eliminar filas

ID 971 · título recuperado de la página española

Sintaxis

```
delrows(Matriz, Entero)  
delrows(Matriz, Ent1..Ent2)
```

Dada una matriz y un entero n , elimina la n -ésima fila de la matriz y devuelve el resultado. Si se utiliza un intervalo de dos enteros en lugar de un único entero, elimina todas las filas en el intervalo y devuelve el resultado.

0972 Lista Delta

ID 972 · título recuperado de la página española

Sintaxis

```
deltalist(Lista)  
deltalist(Vector)
```

Crea una nueva lista o vector formado por las primeras diferencias de una lista o vector; es decir, las diferencias entre los elementos consecutivos en la lista. La lista nueva tiene un elemento menos que la lista original.

0681 Denominador simplificado

ID 681 · título recuperado de la página española

Sintaxis

denom(a/b)

Para los números enteros a y b, devuelve el denominador de la fracción a/b después de la simplificación.

0716 Resolver ecuación diferencial

ID 716 · título recuperado de la página española

Sintaxis

desolve(Ec, [TimeVar], Var)

Devuelve la solución para una ecuación diferencial.

0617 Determinante de la matriz cuadrada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Determinante de la matriz cuadrada.](#)

1682 Número a fracción continua

ID 1682 · título recuperado de la página española

Sintaxis

dfc(real, [entero])
dfc(real, [epsilon])

Devuelve la representación de la fracción continua de real con orden entero utilizando el valor de épsilon de la Configuración de CAS o con epsilon específica.

1683 Fracción continua a número

ID 1683 · título recuperado de la página española

Sintaxis

dfc2f(fracción_continua)

Transforma un vector de fracción_continua en un número real, fracción o valor.

0642 Diagonal

ID 642 · título recuperado de la página española

Sintaxis

diag(list)
diag(matrix)

Si el argumento es una lista, diag devuelve una matriz cuyos elementos diagonales son los elementos de la lista, con ceros en todas las demás posiciones.

Si el argumento es una matriz, diag devuelve un vector que contiene los elementos diagonales de la matriz.

0686 Diferenciar

ID 686 · título recuperado de la página española

Sintaxis

diff(Expr [, Var [, Order]])
diff(Expr [, VarList [, Order]])

El comando diff devuelve la derivada de la expresión especificada por Expr. La derivada se puede tomar con respecto a una variable, especificada por Var, o a una lista de variables, especificada por VarList. También puede

usar la plantilla de diferenciación en el menú Plantillas.

El argumento final opcional `Order` especifica el orden de la derivada a calcular. Se puede usar cuando se proporciona `Var` o `VarList`. Si se omite `Order`, el orden predeterminado es 1.

0973 Dimensiones de la cadena

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dimensiones de la cadena.](#)

0976 Función Delta Dirac

ID 976 · título recuperado de la página española

Sintaxis

`Dirac(Real)`

Devuelve el valor de la función delta de Dirac para un número real.

0199 Distancia

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Distancia.](#)

1012 Distancia al cuadrado

ID 1012 · título recuperado de la página española

Sintaxis

`distance2(Punto1, Punto2)`
`distance2(Punto, Curva)`

Devuelve el cuadrado de la distancia entre dos puntos o entre un punto y una curva.

Ejemplos

[Ejemplo gráfico E1]

Si GA es el punto en (0,0) y GB se define como `plotfunc(4-x2/4)`, entonces la distancia (GA, GB) devuelve 12.

0977 Distancia en

ID 977 · título recuperado de la página española

Sintaxis

`distanceat(Punto1, Punto2, Punto3)`
`distanceat(Punto1, Curva, Punto3)`

Similar al comando `distance`, pero se utiliza en la Vista simbólica de la aplicación Geometría. Muestra la distancia entre dos puntos o entre un punto y una curva y coloca esa medición en la ubicación de `Point3` en la Vista de gráfico. La distancia está etiquetada.

Ejemplos

[Ejemplo gráfico E1] devuelve 2.828... o $2\sqrt{2}$ y coloca esa medida, con una etiqueta, en el origen de la Vista de gráfico.

Si GA es el punto en (0,0) y GB se define como `plotfunc(4-x2/4)`, entonces [Ejemplo gráfico E2] devuelve 3.464... o $2\sqrt{3}$ y coloca esta medida en Vista de gráfico en (0,0).

Define `A:=point(0)` y `B:=point(1+i)`; entonces [Ejemplo gráfico E3] devuelve $\sqrt{2}$ y coloca esta medida en (1/2, 1/2) con una etiqueta.

0693 Divergencia

ID 693 · título recuperado de la página española

Sintaxis

`divergence([Expr1, Expr2, ... ExprN], [Var1, Var2, ... VarN])`

Devuelve la divergencia de un campo vectorial, definido por $\text{divergence}([A,B,C],[x,y,z])=dA/dx+dB/dy+dC/dz$.

0771 Divisores de polinomios

ID 771 · título recuperado de la página española

Sintaxis

```
divis(Poli)
divis({Poli1, Poli2, ...Polin})
```

Dado un polinomio o una lista de polinomios, devuelve un vector que contiene los divisores del polinomio.

1007 Punto de división

ID 1007 · título recuperado de la página española

Sintaxis

```
division_point(PuntoA, PuntoB, Realk)
division_point(CplxA, CplxB, Cplxk)
```

Por dos puntos A y B y un factor numérico k, devuelve un punto C de forma tal que $C - B = k(C - A)$. Se puede hacer referencia a los dos puntos por su nombre o pueden representarse mediante números complejos.

0702 Taylor de cociente

ID 702 · título recuperado de la página española

Sintaxis

```
divpc(Poli1, Poli2, Entero)
```

Devuelve el polinomio de Taylor de n-grados para el cociente de dos polinomios.

Ejemplos

[Ejemplo gráfico E1], el polinomio de quinto grado

1711 Dominio de funciones

ID 1711 · título recuperado de la página española

Sintaxis

```
domain(función [, variable])
```

Devuelve el dominio de variable en función. Si no se asigna, se asume que variable es x.

0658 Producto escalar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Producto escalar.](#)

1009 Dibujar pendiente

ID 1009 · título recuperado de la página española

Sintaxis

```
DrawSlp(a, b, m)
```

Dado tres números reales a, b, y m, dibuja una línea con pendiente m que pasa por el punto (a, b).

1008 Dibujar cadena

ID 1008 · título recuperado de la página española

Sintaxis

```
draw_string(x, y, str [, color=0])
draw_string(str, x, y [, color=0])
```

Dibuja una cadena str en el píxel x, y. Un parámetro color opcional puede especificar el color del texto, proporcionado como un entero de 16 bits en formato RGB565.

0835 Funciones en vista CAS: 'E'

ID 835 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'E'

0982 egcd — Dados dos polinomios, A y B, devuelve tres polinomios U, V y D de forma que: $U(x)*A(x)+V(x)*B(x)=D(x),...$

ID 982 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: egcd

Explicación española: Dados dos polinomios, A y B, devuelve tres polinomios U, V y D de forma que: $U(x)*A(x)+V(x)*B(x)=D(x),...$

Sintaxis

`egcd(PoliA, PoliB, [Var])`
`egcd(ListaA, ListaB, [Var])`

Dados dos polinomios, A y B, devuelve tres polinomios U, V y D de forma que: $U(x)*A(x)+V(x)*B(x)=D(x)$, donde $D(x)=\text{GCD}(A(x),B(x))$, el máximo divisor común de polinomios A y B.

Los polinomios pueden suministrarse en forma simbólica o como listas de coeficientes en orden descendente.

Sin un tercer argumento, se asume que los polinomios son expresiones de x. Con una variable como tercer argumento, los polinomios son expresiones de esa variable.

0670 Integral exponencial

ID 670 · título recuperado de la página española

Sintaxis

`Ei(x)`

Para un valor verdadero x, devuelve el valor aproximado de $\int(e^t)/t, -\infty, x)$

0638 Valores propios

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Valores propios.](#)

0983 Valores propios de una matriz

ID 983 · título recuperado de la página española

Sintaxis

`eigenvals(Matriz)`

Devuelve la secuencia de valores propios de una matriz.

0984 Vectores propios de una matriz

ID 984 · título recuperado de la página española

Sintaxis

`eigenvects(Matriz)`

Devuelve los vectores propios de una matriz diagonalizable.

0639 Valores y vectores propios

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Valores y vectores propios.](#)

0985 eigVc — Devuelve los vectores propios de una matriz diagonalizable.

ID 985 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: eigVc

Explicación española: Devuelve los vectores propios de una matriz diagonalizable.

Sintaxis

eigVc(Matriz)

Devuelve los vectores propios de una matriz diagonalizable.

0986 eigVl — Devuelve la matriz de Jordan asociada con una matriz cuando los valores propios se pueden calcular.

ID 986 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: eigVl

Explicación española: Devuelve la matriz de Jordan asociada con una matriz cuando los valores propios se pueden calcular.

Sintaxis

eigVl(Matriz)

Devuelve la matriz de Jordan asociada con una matriz cuando los valores propios se pueden calcular.

0132 Punto sobre

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Punto sobre.](#)

0167 Elipse

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Elipse.](#)

0166 = (igual)

ID 166 · título recuperado de la página española

Sintaxis

equal(left_expr, right_expr)
equal(var, expr)

Construye una ecuación o define una variable.

equal(left_expr, right_expr) construye una ecuación con left_expr como miembro izquierdo y right_expr como miembro derecho. Si el argumento izquierdo es una variable, equal(var, expr) asigna la expresión expr a la variable var.

La notación funcional equal(A, B) equivale a escribir A=B. Una ecuación A=B es equivalente a A-B=0.

Ejemplos

__E1__
__E2__ asigna el valor 2 a x.

0196 Ecuación de

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ecuación de.](#)

1011 Triángulo equilátero

ID 1011 · título recuperado de la página española

Sintaxis

`equilateral_triangle(Punto1, Punto2, [Var])`

Dibuja un triángulo equilátero definido por uno de sus lados; es decir, por dos vértices consecutivos. El tercer punto se calcula automáticamente, pero no define simbólicamente. Si se agrega una variable en minúsculas como tercer argumento, entonces el tercer punto está etiquetado con el nombre de la variable y las coordenadas del tercer punto se almacenan en esa variable. La orientación del triángulo es hacia la izquierda desde el primer punto.

Ejemplos

[Ejemplo gráfico E1] dibuja el triángulo equilátero a través de los puntos en $(0,0)$, $(1,0)$, y $(1/2, \sqrt{3}/2)$.

0664 Función de error

ID 664 · título recuperado de la página española

Sintaxis

`erf(x)`

Para un valor real x , devuelve el valor aproximado de $2/\sqrt{\pi} \int_0^x e^{-t^2} dt$.

0665 Función de Error complementaria

ID 665 · título recuperado de la página española

Sintaxis

`erfc(x)`

Para un valor real x , devuelve el valor aproximado de $2/\sqrt{\pi} \int_x^\infty e^{-t^2} dt$.

0762 Totiente de Euler

ID 762 · título recuperado de la página española

Sintaxis

`euler(Entero)`

Función phi de Euler (o Totient)

Toma un número entero positivo y devuelve el número de enteros positivos inferiores o iguales a este, que son coprimos.

0993 Constante de Euler-Mascheroni

ID 993 · título recuperado de la página española

Sintaxis

`euler_gamma`

La constante matemática γ (constante de Euler-Mascheroni), representada internamente como 0.577215664902.

0989 eval — Úa una expresión.

ID 989 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `eval`

Explicación española: Úa una expresión.

Sintaxis

`eval(Expr)`

Evalúa una expresión.

1693 Evaluar Booleano

ID 1693 · título recuperado de la página española

Sintaxis

evalb(expresión)

Esta función evaluará expresión a un valor verdadero o falso si es posible.

0990 Evaluar expresión compleja

ID 990 · título recuperado de la página española

Sintaxis

evalc(expresión)

Devuelve una expresión compleja escrita en la forma real + i*imag.

0991 Evaluar expresión numérica

ID 991 · título recuperado de la página española

Sintaxis

evalf(Expr, [Entero])

Dada una expresión y un número de dígitos significativos, devuelve la evaluación numérica de la expresión para el número de dígitos significativos determinado. Con solo una expresión, devuelve la evaluación numérica basada en la configuración de CAS.

1003 Prueba de paridad

ID 1003 · título recuperado de la página española

Sintaxis

even(Entero)

Pruebas si un número entero es o no es par. Devuelve 1 si es y 0 si no lo es.

0992 Conversión exacta

ID 992 · título recuperado de la página española

Sintaxis

exact(Expr)

Convierte una expresión decimal en una expresión racional o real dentro de la tolerancia de Epsilon especificada en la configuración de CAS.

1010 Función exbisector

ID 1010 · título recuperado de la página española

Sintaxis

exbisector(Punto1, Punto2, Punto3)

Dados tres puntos que definen un triángulo, crea el bisector de los ángulos exteriores del triángulo cuyo vértice común se encuentra en el primer punto. No es necesario dibujar el triángulo en la Vista de gráfico.

Ejemplos

[Ejemplo gráfico E1] dibuja el bisector de los ángulos exteriores de $\triangle ABC$ cuyo vértice común se encuentra en el punto A.
[Ejemplo gráfico E2] dibuja la línea dada por $y=x$

0163 Excírculo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Excírculo.](#)

0995 Exponencial natural

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Exponencial natural.](#)

0724 exp2pow — Transforma una expresión de la forma $e^{(n \cdot \ln(x))}$ reescrita como una potencia de x.

ID 724 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: exp2pow

Explicación española: Transforma una expresión de la forma $e^{(n \cdot \ln(x))}$ reescrita como una potencia de x.

Sintaxis

exp2pow(Expr)

Transforma una expresión de la forma $e^{(n \cdot \ln(x))}$ reescrita como una potencia de x. Aplica $e^{(n \cdot \ln(x))} = x^n$.

0726 exp2trig — Devuelve una expresión con complejos exponenciales reescritos en términos de seno y coseno.

ID 726 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: exp2trig

Explicación española: Devuelve una expresión con complejos exponenciales reescritos en términos de seno y coseno.

Sintaxis

exp2trig(Expr)

Devuelve una expresión con complejos exponenciales reescritos en términos de seno y coseno.

0676 Ampliar expresión

ID 676 · título recuperado de la página española

Sintaxis

expand(Expr)

Devuelve una expresión ampliada.

0727 Ampliar exponenciales

ID 727 · título recuperado de la página española

Sintaxis

expexpand(Expr)

Amplía los exponenciales utilizando la identidad $e^{(a \cdot f(x))} = e^{f(x)}^a$.

0551 Exponente menos 1

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Exponente menos 1.](#)

0996 Distribución exponencial

ID 996 · título recuperado de la página española

Sintaxis

exponential(k, x)

Función de densidad de probabilidad exponencial

Calcula la densidad de probabilidad de la distribución exponencial en x, con el parámetro k.

0997 Exponencial acumulada

ID 997 · título recuperado de la página española

Sintaxis

```
exponential_cdf(k, x, [x2])
```

Función de distribución exponencial acumulada

Devuelve la probabilidad de cola inferior de la función de densidad de probabilidad exponencial para el valor x , con el parámetro k .

0998 Exponencial acumulada inversa

ID 998 · título recuperado de la página española

Sintaxis

```
exponential_icdf(k, p)
```

Función de distribución de exponencial acumulada inversa

Devuelve el valor de x de forma tal que la probabilidad de cola inferior exponencial de x , con el parámetro k , sea p .

0999 Regresión exponencial

ID 999 · título recuperado de la página española

Sintaxis

```
exponential_regression(Matriz)
exponential_regression(Lista1, Lista2)
exponential_regression(Vector1, Vector2)
```

Dado un conjunto de puntos, devuelve un vector que contiene los coeficientes a y b de $y=b*a^x$, el exponencial que mejor se adapta al conjunto de puntos. Los puntos pueden ser los elementos de dos listas o las filas de una matriz $2 \times n$.

1002 Evaluar cadena

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Evaluar cadena.](#)

1013 Extraer medición

ID 1013 · título recuperado de la página española

Sintaxis

```
extract_measure(Var)
```

Devuelve la definición de un objeto geométrico. Para un punto, esa definición consiste en las coordenadas del punto. Para otros objetos, la definición refleja su definición en la Vista simbólica, con las coordenadas de los puntos de definición suministrados.

1004 extend — Agrega elementos de `Source` al final de `Destination`.

ID 1004 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: extend

Explicación española: Agrega elementos de `Source` al final de `Destination`.

Sintaxis

```
extend(Destination, Source)
```

Agrega elementos de `Source` al final de `Destination`. Tanto `Source` como `Destination` pueden ser listas o vectores.

1005 ezgcd — Utiliza el algoritmo de EZ GCD para devolver el máximo común divisor de dos polinomios con al menos dos...

ID 1005 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ezgcd

Explicación española: Utiliza el algoritmo de EZ GCD para devolver el máximo común divisor de dos polinomios con al menos dos...

Sintaxis

ezgcd(Poli1, Poli2)

Utiliza el algoritmo de EZ GCD para devolver el máximo común divisor de dos polinomios con al menos dos variables.

0836 Funciones en vista CAS: 'F'

ID 836 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'F'

1020 Fracción → Numerador/Denominador

ID 1020 · título recuperado de la página española

Sintaxis

f2nd(Frac)

f2nd(RatFrac)

Devuelve un vector compuesto por el numerador y el denominador de una forma irreducible de una fracción racional.

0677 Factorizar polinomio

ID 677 · título recuperado de la página española

Sintaxis

factor(Expr)

Devuelve un polinomio factorizado

Similar a recopilar, pero factorizará usando raíces cuadradas.

0785 Factor por grado

ID 785 · título recuperado de la página española

Sintaxis

factor_xn(Poli)

Para un polinomio específico en x de grado n, factoriza x^n y devuelve el producto resultante.

1021 Factorial

ID 1021 · título recuperado de la página española

Sintaxis

factorial(Entero)

factorial(Real)

Devuelve el factorial de un entero o la solución a la función gamma para un número no entero. Para un número entero n, $\text{factorial}(n)=n!$. Para un número real no entero a, $\text{factorial}(a)=a! = \text{Gamma}(a + 1)$.

0772 Lista de factor polinomial

ID 772 · título recuperado de la página española

Sintaxis

```
factors(Poli)
factors({Poli1, Poli2, ..., Polin})
```

Devuelve la lista de los principales factores de un polinomio; cada factor seguido por su multiplicidad.

0778 Raíces a polinomio

ID 778 · título recuperado de la página española

Sintaxis

```
fcoeff([Raíz1, Orden1, Raíz2, Orden2, ..., RaízN, OrdenN])
```

Devuelve el polinomio que se describe mediante una lista de raíces, cada uno seguido por su orden.

0707 Transformada rápida de Fourier

ID 707 · título recuperado de la página española

Sintaxis

```
fft(Vector)
fft(Vector, a, p)
```

Con un argumento (un vector), devuelve la transformada de Fourier discreta en R.

Con dos argumentos de número entero adicionales a y p, devuelve la transformada de Fourier discreta en el campo $\mathbb{Z}/p\mathbb{Z}$, con una raíz n-ésima primitiva de 1 ($n=\text{size}(\text{Vector})$).

0581 Densidad de Fisher

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Densidad de Fisher.](#)

0589 Fisher acumulada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Fisher acumulada.](#)

0597 Fisher acumulada inversa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Fisher acumulada inversa.](#)

0598 Conversión a número de punto flotante

ID 598 · título recuperado de la página española

Sintaxis

```
float(Obj)
```

Convierte una entrada dada a un número de punto flotante. El objeto `Obj` puede ser cualquier número, una cadena, una lista de números o un vector de números. Si se proporciona una cadena, debe ser una representación válida de un número, que puede incluir espacios al principio y al final, los cuales se ignoran. La función también entiende cadenas que representan valores especiales de punto flotante como " ∞ " o "infinity".

0530 Suelo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Suelo.](#)

1022 Función máxima

ID 1022 · título recuperado de la página española

Sintaxis

fMax(Expr, [Var])

Dada una expresión en x, devuelve el valor de x para el cual la expresión tiene su valor máximo. Dada una expresión y una variable, devuelve el valor de esa variable para la que la expresión tiene su valor máximo.

1023 Función mínima

ID 1023 · título recuperado de la página española

Sintaxis

fMin(Expr, [Var])

Dada una expresión en x, devuelve el valor de x para el cual la expresión tiene su valor mínimo. Dada una expresión y una variable, devuelve el valor de esa variable para la que la expresión tiene su valor mínimo.

0541 Encontrar raíz

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Encontrar raíz.](#)

1028 Número de formato

ID 1028 · título recuperado de la página española

Sintaxis

format(real, formato)

Devuelve el número real como una cadena en format (f=flotante, s=científico, e=ingeniería, a=hexadecimal).

1024 **fourier_an** — Devuelve el *n*ésimo coeficiente de Fourier $an=2/T \int (f(x) \cos(2\pi n x/T), a, a+T)$.

ID 1024 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: fourier_an

Explicación española: Devuelve el *n*ésimo coeficiente de Fourier $an=2/T \int (f(x) \cos(2\pi n x/T), a, a+T)$.

Sintaxis

fourier_an(Expr, Var, T, n, a)

Devuelve el *n*ésimo coeficiente de Fourier $an=2/T \int (f(x) \cos(2\pi n x/T), a, a+T)$.

1025 **fourier_bn** — Devuelve el *n*ésimo coeficiente de Fourier $bn=2/T \int (f(x) \sin(2\pi n x/T), a, a+T)$.

ID 1025 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: fourier_bn

Explicación española: Devuelve el *n*ésimo coeficiente de Fourier $bn=2/T \int (f(x) \sin(2\pi n x/T), a, a+T)$.

Sintaxis

fourier_bn(Expr, Var, T, n, a)

Devuelve el *n*ésimo coeficiente de Fourier $bn=2/T \int (f(x) \sin(2\pi n x/T), a, a+T)$.

1026 **fourier_cn** — Devuelve el *n*ésimo coeficiente de Fourier $cn=1/T \int (f(x) \exp(-2\pi i n x/T), a, a+T)$.

ID 1026 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: fourier_cn

Explicación española: Devuelve el *n*ésimo coeficiente de Fourier $cn=1/T \int (f(x) \exp(-2\pi i n x/T), a, a+T)$.

Sintaxis

```
fourier_cn(Expr, Var, T, n, a)
```

Devuelve el n -ésimo coeficiente de Fourier $c_n = 1/T \int (f(x) \cdot \exp(-2\pi i n x / T)) dx$, $a, a+T$.

1029 Parte fraccional

ID 1029 · título recuperado de la página española

Sintaxis

```
fPart(Real)
fPart(Lista)
```

Devuelve la parte fraccional de un número real o las partes fraccionales de una lista de números reales.

1030 fracmod — Dado un entero n y un módulo p , **fracmod** encuentra la fracción igual a $n \pmod{p}$, donde tanto el...

ID 1030 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `fracmod`

Explicación española: Dado un entero n y un módulo p , **fracmod** encuentra la fracción igual a $n \pmod{p}$, donde tanto el...

Sintaxis

```
fracmod(n, p)
```

Dado un entero n y un módulo p , **fracmod** encuentra la fracción igual a $n \pmod{p}$, donde tanto el numerador como el denominador están restringidos a valores menores o iguales a $\sqrt{p}/2$ en valor absoluto.

Parámetros

- n : un entero (que representa una fracción).
- p : un entero (el módulo).

fracmod(n, p) devuelve, si es posible, una fracción a/b tal que:

- $a \equiv n \pmod{p}$
- $-\sqrt{p}/2 < a \leq \sqrt{p}/2$
- $0 \leq b < \sqrt{p}/2$

1032 Extracción de mantisa y exponente

ID 1032 · título recuperado de la página española

Sintaxis

```
fexp(Num)
```

Para un número de punto flotante dado Num , devuelve un vector de dos valores [mantisa, exponente], donde Num se representa como: mantisa $\cdot 2^{\text{exponente}}$ (exactamente).

1033 Raíces y polos numéricos

ID 1033 · título recuperado de la página española

Sintaxis

```
froot(RatPoly)
```

Devuelve la lista de raíces y polos de un polinomio racional con sus multiplicidades.

0715 Solución numérica

ID 715 · título recuperado de la página española

Sintaxis

```
fsolve(Expr [, Var [, Init [, Method]])  
fsolve(ExprVector, VersVector [, Init [, Method]])
```

Devuelve la solución numérica de una ecuación o un sistema de ecuaciones.

Si Var no se especifica, se asume la variable predeterminada 'x'.

Con el argumento opcional Init, puede especificar una suposición para la solución o un intervalo dentro del cual se espera que ocurra la solución.

Con el argumento opcional Method, puede especificar el nombre del algoritmo iterativo que utilizará el solucionador. Si está resolviendo para una sola variable, sus opciones son bisection_solver, brent_solver, falsepos_solver, newton_solver, secant_solver o steffenson_solver. Si está resolviendo para 2 variables, su opción es solo newton_solver.

1034 Derivada funcional

ID 1034 · título recuperado de la página española

Sintaxis

```
function_diff(NombreFunción)
```

Devuelve la función derivada como una asignación de x en la derivada de la función especificada.

0837 Funciones en vista CAS: 'G'

ID 837 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'G'

0666 Función Gamma

ID 666 · título recuperado de la página española

Sintaxis

```
Gamma(Real(n), [Real(a)])  
Gamma(Complex(n))
```

Con un (real o complejo) argumento, devuelve el valor de la función gamma (completa) $\Gamma(n) = \Gamma(n,0)$

Con dos (reales) argumentos, devuelve la función gamma (incompleta) $\Gamma(n,a)$.

$\Gamma(n) = (n-1)!$ si n es un entero

1035 Densidad Gamma

ID 1035 · título recuperado de la página española

Sintaxis

```
gammad(a, t, x)
```

Función de densidad de probabilidad Gamma

Calcula la densidad de probabilidad de la distribución gamma a x, con los parámetros a y t.

1036 Gamma acumulada

ID 1036 · título recuperado de la página española

Sintaxis

```
gammad_cdf(a, t, x, [x2])
```

Función de distribución de gama acumulada

Devuelve la probabilidad de cola más baja de la función de densidad de la probabilidad gamma para el valor x, dados los parámetros a y t. Con el cuarto argumento opcional x2, devuelve el área entre los dos valores de x.

1037 Gamma acumulada inversa

ID 1037 · título recuperado de la página española

Sintaxis

`gammad_icdf(a, t, p)`

Función de distribución de gamma acumulada inversa

Devuelve el valor de x de forma tal que la probabilidad de la cola inferior gamma de x , con los parámetros a y t , sea p .

1038 gauss — Dada una expresión seguida por un vector de variables, utiliza el algoritmo de Gauss para devolver la...

ID 1038 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `gauss`

Explicación española: Dada una expresión seguida por un vector de variables, utiliza el algoritmo de Gauss para devolver la...

Sintaxis

`gauss(Expr, Vector)`

Dada una expresión seguida por un vector de variables, utiliza el algoritmo de Gauss para devolver la forma cuadrática de la expresión escrita como una suma o diferencia de cuadrados de las variables dadas en el vector.

0791 Bases de Groebner

ID 791 · título recuperado de la página española

Sintaxis

`gbasis([Poli1, Poli2, ...], [Var1, Var2, ...])`

Dado un vector de polinomios y un vector de variables, devuelve las bases de Groebner del ideal abarcado por el conjunto de polinomios.

0773 Divisor común máximo

ID 773 · título recuperado de la página española

Sintaxis

`gcd(Poly_1, Poly_2 [, ..., Poly_N])`
`gcd(Int_1, Int_2 [, ..., Int_N])`

Devuelve el divisor común máximo de dos o más polinomios o el divisor común máximo de dos o más enteros.

0799 Divisor común máximo

ID 799 · título recuperado de la página española

Sintaxis

`gcd(Poli1, Poli2)`
`gcd(Entero1, Entero2)`

Devuelve el divisor común máximo de 2 polinomios de varias variables. También puede ser utilizado como `gcd` entero.

0583 Densidad geométrica

ID 583 · título recuperado de la página española

Sintaxis

`GEOMETRIC(p, x)`

Función de densidad de probabilidad geométrica

Calcula la densidad de probabilidad de la distribución geométrica en x , con la probabilidad p .

0591 Geometría acumulada

ID 591 · título recuperado de la página española

Sintaxis

`geometric_cdf(p, x, [x2])`

Función de distribución geométrica acumulada

Con dos valores (p y x), devuelve la probabilidad de cola inferior de la función de densidad de probabilidad geométrica para el valor x , con la probabilidad p . Con tres valores (p , x_1 y x_2), devuelve el área bajo la función de densidad de probabilidad geométrica definida por la probabilidad p , entre x_1 y x_2 .

0600 Geometría acumulada inversa

ID 600 · título recuperado de la página española

Sintaxis

`geometric_icdf(p, k)`

Función de distribución geométrica acumulada inversa

Devuelve el valor x que tiene el valor de probabilidad de cola inferior k , con la probabilidad p .

1044 Crear campo de Galois

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Crear campo de Galois.](#)

0694 Gradiente

ID 694 · título recuperado de la página española

Sintaxis

`grad(Expr, ListVars)`

Devuelve el gradiente de una expresión.

Con una lista de variables como segundo argumento, devuelve el vector de derivadas parciales.

1045 Ortonormalización de Gramschmidt

ID 1045 · título recuperado de la página española

Sintaxis

`gramschmidt(Vector, Función)`

Dada una base de un subespacio vectorial y una función que define un producto escalar en este subespacio vectorial, devuelve una base ortonormal para esa función.

0792 Groebner Remainder (Resto de Groebner)

ID 792 · título recuperado de la página española

Sintaxis

`greduce(Poli1, [Poli2, Poli3, ...], [Var1, Var2, ...])`

Dado un polinomio, un vector de polinomios y un vector de variables, devuelve el resto de la división del polinomio por la base de Groebner del vector de polinomios.

1707 groupermu — Devuelve el grupo de permutaciones generadas por permutación1 y permutación2.

ID 1707 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: groupermu

Explicación española: Devuelve el grupo de permutaciones generadas por permutación1 y permutación2.

Sintaxis

`groupermu(permutación1, permutación2)`

Devuelve el grupo de permutaciones generadas por permutación1 y permutación2.

0838 Funciones en vista CAS: 'H'

ID 838 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'H'

1050 hadamard — Límite Hadamard de una matriz o elemento mediante multiplicación del elemento de 2 matrices.

ID 1050 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: hadamard

Explicación española: Límite Hadamard de una matriz o elemento mediante multiplicación del elemento de 2 matrices.

Sintaxis

`hadamard(Matriz, [Matriz])`

Límite Hadamard de una matriz o elemento mediante multiplicación del elemento de 2 matrices.

0142 Raya

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Raya.](#)

0740 halftan — Transforma sin(x), cos(x) y tan(x) como una función de tan(x/2).

ID 740 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: halftan

Explicación española: Transforma sin(x), cos(x) y tan(x) como una función de tan(x/2).

Sintaxis

`halftan(Expr)`

Transforma sin(x), cos(x) y tan(x) como una función de tan(x/2).

1051 halftan_hyp2exp — Transforma las funciones de trigonometría en tan(x/2) y las funciones hiperbólicas en exponenciales.

ID 1051 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: halftan_hyp2exp

Explicación española: Transforma las funciones de trigonometría en tan(x/2) y las funciones hiperbólicas en exponenciales.

Sintaxis

`halftan_hyp2exp(Expr)`

Transforma las funciones de trigonometría en tan(x/2) y las funciones hiperbólicas en exponenciales.

1052 halt — Pone un programa en modo de depuración paso a paso.

ID 1052 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: halt

Explicación española: Pone un programa en modo de depuración paso a paso.

Sintaxis

halt

Pone un programa en modo de depuración paso a paso.

1053 Distancia de Hamming

ID 1053 · título recuperado de la página española

Sintaxis

hamdist(Ent1, Ent2)

Devuelve la distancia de Hamming entre dos números enteros.

1054 Conjugado armónico

ID 1054 · título recuperado de la página española

Sintaxis

harmonic_conjugate(Punto1, Punto2, Punto3)

harmonic_conjugate(Línea1, Línea2, Línea3)

Devuelve el conjugado armónico de 3 puntos. Específicamente, devuelve el conjugado armónico de Punto3 con respecto a Punto1 y Punto2. También acepta tres líneas paralelas o concurrentes; en este caso, devuelve la ecuación de la línea conjugada armónica.

1055 División armónica

ID 1055 · título recuperado de la página española

Sintaxis

harmonic_division(punto1, punto2, punto3, var)

harmonic_division(línea1, línea2, línea3, var)

Devuelve los 4 puntos (líneas resp) y afecta al último argumento, de forma que los 4 puntos (líneas resp) están en una división armónica.

1056 Variable Has

ID 1056 · título recuperado de la página española

Sintaxis

has(Expr, Var)

Comprueba si hay una variable en una expresión.

Devuelve 1 si la variable está en la expresión; de lo contrario devuelve 0.

1057 Elemento Head

ID 1057 · título recuperado de la página española

Sintaxis

head(Vector)

head(String)

head(Obj1 [, Obj2, ...])

Devuelve el primer elemento de un vector o una cadena o un conjunto de objetos.

1058 Heaviside — Devuelve el valor de la función Heaviside para un número real determinado.

ID 1058 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Heaviside

Explicación española: Devuelve el valor de la función Heaviside para un número real determinado.

Sintaxis

Heaviside(Real)

Devuelve el valor de la función Heaviside para un número real determinado.

La función de Heaviside es igual a 0 si $x < 0$ y 1 si $x \geq 0$.

0793 Polinomio de Hermite

ID 793 · título recuperado de la página española

Sintaxis

hermite(Entero)

Devuelve el polinomio de Hermite de grado n , donde n es un número entero menor a 1556.

0645 Hessenberg

ID 645 · título recuperado de la página española

Sintaxis

hessenberg(Matriz_A)

Dado Matriz A, devuelve la reducción de la matriz a la forma de Hessenberg. Devuelve [P,B] de forma que $B = \text{inv}(P) * A * P$.

0695 Matriz hessiana

ID 695 · título recuperado de la página española

Sintaxis

hessian(Expr, ListVar)

Devuelve la matriz hessiana de una expresión.

0915 Entero a cadena hexadecimal

ID 915 · título recuperado de la página española

Sintaxis

hex(Integer)

Convierte un entero en una cadena que representa la forma hexadecimal en minúsculas del argumento Integer, precedida por "0x".

1093 Hexágono

ID 1093 · título recuperado de la página española

Sintaxis

hexagon(Punto1, Punto2, [Var1, Var2, Var3, Var4])

Dibuja un hexágono regular definido por uno de sus lados; es decir, por dos vértices consecutivos. Los puntos restantes se calculan automáticamente, pero no se definen simbólicamente. La orientación del hexágono es hacia la izquierda desde el primer punto.

Ejemplos

[Ejemplo gráfico E1] dibuja un hexágono regular, cuyos dos primeros vértices están en (0, 0) y (6, 0).

[Ejemplo gráfico E2] dibuja un hexágono regular, cuyos dos primeros vértices están en (0, 0) y (6, 0)l etiqueta los otros cuatro vértices a, b, c y d, y almacena las coordenadas en las variables de CAS a, b, c y d.

No tiene que definir las variables para todos los cuatro puntos restantes, pero las coordenadas se almacenan en orden. Por ejemplo, [Ejemplo gráfico E3] almacena solo el tercer punto en la variable de CAS a.

0624 Matriz de Hilbert

ID 624 · título recuperado de la página española

Sintaxis

hilbert(n)

Dado un número entero positivo n, devuelve la matriz de Hilbert de orden n-ésimo. Cada elemento de la matriz se proporciona en la fórmula $1/(j+k-1)$ donde j es el número de fila y k es el número de columna.

0186 Dilación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dilación.](#)

1061 horner — Devuelve el valor de un polinomio P(a) calculado con el método de Horner.

ID 1061 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: horner

Explicación española: Devuelve el valor de un polinomio P(a) calculado con el método de Horner.

Sintaxis

horner(Polinómica, Real)

Devuelve el valor de un polinomio P(a) calculado con el método de Horner. El polinomio puede darse como una expresión simbólica o como un vector de coeficientes.

1062 hyp2exp — Devuelve una expresión con los términos hiperbólicos reescritos como exponenciales.

ID 1062 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: hyp2exp

Explicación española: Devuelve una expresión con los términos hiperbólicos reescritos como exponenciales.

Sintaxis

hyp2exp(Expr)

Devuelve una expresión con los términos hiperbólicos reescritos como exponenciales.

0168 Hipérbola

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Hipérbola.](#)

0839 Funciones en vista CAS: 'I'

ID 839 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'I'

1063 iabcuv — Devuelve [u,v] como $au+bv=c$ para los 3 números enteros a,b,c

ID 1063 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: iabcuv

Explicación española: Devuelve [u,v] como $au+bv=c$ para los 3 números enteros a,b,c

Sintaxis

```
iabcuv(Ent(a), Ent(b), Ent(C))
```

Devuelve $[u,v]$ como $au+bv=c$ para los 3 números enteros a,b,c

1064 Base de intersección

ID 1064 · título recuperado de la página española

Sintaxis

```
ibasis(Matriz1, Matriz2)
```

Dadas dos matrices, las interpreta como dos espacios vectoriales y devuelve la base vectorial de su intersección.

0697 Integración por partes v

ID 697 · título recuperado de la página española

Sintaxis

```
ibpdv(f(Var), v(Var) [, Var [, Real1, Real2]])
```

Realiza integración por partes de la expresión $f(x)=u(x)*v'(x)$, con $f(x)$ como el primer argumento y $v(x)$ (o 0) como el segundo argumento.

Específicamente, devuelve un vector cuyo primer elemento es $u(x)*v(x)$ y cuyo segundo elemento es $v(x)*u'(x)$. Con el tercer, cuarto y quinto argumento opcional, puede especificar una variable de integración y los límites de la integración. Si no se proporciona ninguna variable de integración, se toma como x .

0698 Integración por partes u

ID 698 · título recuperado de la página española

Sintaxis

```
ibpu(f(Var), u(Var) [, Var [, Real1, Real2]])
```

Realiza la integración por partes de la expresión $f(x)=u(x)*v'(x)$ con $f(x)$ como el primer argumento y $u(x)$ (o 0) como segundo argumento.

Específicamente, devuelve un vector cuyo primer elemento es $u(x)*v(x)$ y cuyo segundo elemento es $v(x)*u'(x)$. Con el tercer, cuarto y quinto argumento opcional, puede especificar una variable de integración y los límites de la integración. Si no se proporciona ninguna variable de integración, se toma como x .

0767 Resto chino con enteros

ID 767 · título recuperado de la página española

Sintaxis

```
ichinrem([a, p], [b, q])
```

Teorema chino del resto con enteros para dos ecuaciones. Toma dos listas, $[a, p]$ y $[b, q]$ y devuelve una lista de dos enteros, $[r, n]$, de tal manera $x \equiv r \pmod n$. En este caso, x es $t \pmod p$ y $x \equiv b \pmod q$. Además, $n=p*q$.

1065 GCD de coeficientes enteros

ID 1065 · título recuperado de la página española

Sintaxis

```
icontent(Poli, [Var])
```

Devuelve el máximo común divisor de los coeficientes enteros de un polinomio.

1066 Función de identidad

ID 1066 · título recuperado de la página española

Sintaxis

```
id(Expr)
```

La función de identidad de entidad: $x \rightarrow x$. Devuelve un conjunto que contiene el argumento original.

0621 Matriz de identidad**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Matriz de identidad.](#)

1067 Matriz de identidad

ID 1067 · título recuperado de la página española

Sintaxis

```
identity(Entero)
```

Dado un número entero n , devuelve la matriz de identidad de dimensión n .

0752 Divisores de enteros

ID 752 · título recuperado de la página española

Sintaxis

```
idivis(Integer)
idivis(Lista)
```

Para una entrada de un solo entero, devuelve un vector que contiene todos sus divisores.

Para una lista de enteros, devuelve una lista de vectores. Cada vector en la lista contiene los divisores del entero correspondiente.

1068 MDC extendido de entero

ID 1068 · título recuperado de la página española

Sintaxis

```
iegcd(Entero1, Entero2)
```

Dados dos enteros a y b , devuelve el máximo divisor común extendido de dos enteros. Devuelve $[u, v, \text{igcd}(a, b)]$ de forma que $a \cdot u + b \cdot v = \text{igcd}(a, b)$.

0753 Factores de entero

ID 753 · título recuperado de la página española

Sintaxis

```
ifactor(Entero)
```

Devuelve la lista de factores primos de un entero como un producto.

Se puede utilizar con `Almca`.

Nota: en algunos casos, la factorización puede fallar. En estos casos, el comando devolverá el producto de -1 y lo opuesto a la entrada original. El -1 indica que la factorización falló.

0754 Lista de factores de enteros

ID 754 · título recuperado de la página española

Sintaxis

```
ifactors(Entero)
```

Similar a `ifactor`, pero devuelve una lista de los factores de los enteros con sus multiplicidades.

0708 Transformada de Fourier rápida Inversa

ID 708 · título recuperado de la página española

Sintaxis

`ifft(Vect)`

Devuelve la transformada inversa de Fourier discreta.

0755 DCM de entero

ID 755 · título recuperado de la página española

Sintaxis

`igcd(Int1, Int2 [, ..., IntN])`

Devuelve el número entero que es el divisor común máximo de dos o más enteros.

0644 Normal de Hermite

ID 644 · título recuperado de la página española

Sintaxis

`ihermite(Matriz_A)`

Dado Matriz A, devuelve la forma normal de Hermite de una matriz con los coeficientes en Z: devuelve U, B de forma que U es invertible en Z, B es la figura triangular superior y $B=U*A$

1072 Transformada de Laplace inversa

ID 1072 · título recuperado de la página española

Sintaxis

`ilaplace(Expr [, Var [, IlapVar]])`

Devuelve la transformada de Laplace inversa de una fracción racional.

0546 Parte imaginaria

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Parte imaginaria.](#)

1073 image — N de una aplicación lineal de una matriz.

ID 1073 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `image`

Explicación española: N de una aplicación lineal de una matriz.

Sintaxis

`image(Matriz)`

Imagen de una aplicación lineal de una matriz.

1074 Derivada implícita

ID 1074 · título recuperado de la página española

Sintaxis

`implicit_diff(Expr, Var1, Var2 [, Order])`

El comando `implicit_diff` devuelve la derivada implícita de la expresión o ecuación especificada por `Expr`. El resultado es una expresión que define la derivada de la variable `Var2` con respecto a la variable `Var1`, por lo que el orden de las variables `Var1` y `Var2` es importante.

El argumento `Expr` suele ser una ecuación. Si `Expr` no contiene un signo igual, se asume la ecuación $Expr = 0$.

El argumento final opcional `Order` especifica el orden de la derivada a calcular. Si se omite `Order`, el orden por defecto es 1.

0164 Incírculo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Incírculo.](#)

0165 index — Para una cadena, devuelve el índice de la primera aparición de `Phrase` en el `String`.

ID 165 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `index`

Explicación española: Para una cadena, devuelve el índice de la primera aparición de ``Phrase`` en el ``String``.

Sintaxis

```
index(String, Phrase)
index(List, Element)
index(Vector, Element)
```

Para una cadena, devuelve el índice de la primera aparición de `Phrase` en el `String`.

Para una lista o vector, devuelve el índice de la primera aparición de `Element` en la `List` o `Vector`.

1077 En cadena

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de En cadena.](#)

0687 Integrar

ID 687 · título recuperado de la página española

Sintaxis

```
int(Expr [, Var [, Real1, Real2]])
```

Devuelve la integral de una expresión.

Con una expresión como argumento, devuelve la integral indefinida con respecto a `x`. Con el segundo, terceros y cuarto argumento opcional, puede especificar la variable de integración y los límites de un integral definida.

0137 Intersecciones

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Intersecciones.](#)

1078 interval2center — Devuelve el centro de un intervalo u objeto.

ID 1078 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `interval2center`

Explicación española: Devuelve el centro de un intervalo u objeto.

Sintaxis

```
interval2center(Intervalo)
interval2center(Objeto)
```

Devuelve el centro de un intervalo u objeto.

1079 Inverso

ID 1079 · título recuperado de la página española

Sintaxis

```
inv(Expr)  
inv(Matriz)
```

Devuelve la inversa de una expresión o una matriz.

0189 Inversión**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Inversión.](#)

0706 Transformada de Laplace inversa

ID 706 · título recuperado de la página española

Sintaxis

```
invlaplace(Expr, [Var], [IlapVar])
```

Devuelve la transformada inversa de Laplace de una expresión.

1082 Transformada Z inversa

ID 1082 · título recuperado de la página española

Sintaxis

```
invztrans(Expr, [Var], [InvZtransVar])
```

Devuelve la transformada z inversa de una fracción racional.

0531 Parte entera**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Parte entera.](#)

1083 Parte entera

ID 1083 · título recuperado de la página española

Sintaxis

```
iPart(Real)  
iPart(Lista)
```

Devuelve un número real sin su parte fraccionaria o una lista de números reales sin su parte fraccionaria.

0764 Cociente euclidiano de entero

ID 764 · título recuperado de la página española

Sintaxis

```
iquo(Ent1, Ent2)
```

Devuelve el cociente entero de la división euclídea de dos números enteros.

1084 Cociente y resto euclidianos

ID 1084 · título recuperado de la página española

Sintaxis

```
iquorem(Entero1, Entero2)
```

Devuelve el cociente euclidiano y el resto de dos números enteros.

0765 Resto euclidiano entero

ID 765 · título recuperado de la página española

Sintaxis

`irem(Ent1, Ent2)`

Devuelve el resto entero de la división euclídea de dos números enteros.

0207 Función `is_collinear`

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función `is\_collinear`.](#)

0208 Función `is_concyclic`

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función `is\_concyclic`.](#)

0215 Función `is_conjugate`

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función `is\_conjugate`.](#)

1094 Función `is_coplanar`

ID 1094 · título recuperado de la página española

Sintaxis

`is_coplanar(Punto1, Punto2, Punto3, Punto4)`

Analiza si cuatro puntos están en el mismo plano.

Retorna 1 si es cierto o 0 si es falso.

1699 Función `is_cycle`

ID 1699 · título recuperado de la página española

Sintaxis

`is_cycle(lista)`

Prueba si lista es un ciclo o no. Devuelve 1 si es afirmativo, y 0 de lo contrario.

0209 Función `is_element`

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función `is\_element`.](#)

0213 Función `is_equilateral`

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función `is\_equilateral`.](#)

1085 `is_harmonic` — Prueba si 4 puntos están o no en una división o rango armónico.

ID 1085 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `is_harmonic`

Explicación española: Prueba si 4 puntos están o no en una división o rango armónico.

Sintaxis

`is_harmonic(Punto1, Punto2, Punto3, Punto4)`

Prueba si 4 puntos están o no en una división o rango armónico.

Devuelve 1 si lo están, de lo contrario devuelve 0.

1086 is_harmonic_circle_bundle — Devuelve 1 si los círculos tienen una tangente externa común 2 si tienen el mismo centro 3 si son el...

ID 1086 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `is_harmonic_circle_bundle`

Explicación española: Devuelve 1 si los círculos tienen una tangente externa común 2 si tienen el mismo centro 3 si son el...

Sintaxis

`is_harmonic_circle_bundle(Círculo1, Círculo2, ..., Círculon)`

Devuelve

1 si los círculos tienen una tangente externa común

2 si tienen el mismo centro

3 si son el mismo círculo

0 si no aplica ninguna de las anteriores

1087 is_harmonic_line_bundle — Devuelve: 1 si las líneas tienen un punto común 2 si todas son paralelas 3 si son la misma línea de lo...

ID 1087 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `is_harmonic_line_bundle`

Explicación española: Devuelve: 1 si las líneas tienen un punto común 2 si todas son paralelas 3 si son la misma línea de lo...

Sintaxis

`is_harmonic_line_bundle(Línea1, Línea2, ..., LíneaN)`

Devuelve:

1 si las líneas tienen un punto común

2 si todas son paralelas

3 si son la misma línea

de lo contrario, devuelve 0

0212 Función is_isosceles

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función is_isosceles.](#)

1095 Función is_orthogonal

ID 1095 · título recuperado de la página española

Sintaxis

`is_orthogonal(Línea1, Línea2)`

`is_orthogonal(Círculo1, Círculo2)`

Prueba si dos líneas o círculos son ortogonales o no (perpendicular). En el caso de dos círculos, prueba si las líneas tangentes en un punto de la intersección son ortogonales o no. Devuelve 1 si lo son; de lo contrario, devuelve 0.

0210 Función is_parallel

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función is_parallel.](#)

0214 Función is_parallelogram

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función is_parallelogram.](#)

1698 Función is_permu

ID 1698 · título recuperado de la página española

Sintaxis

is_permu(lista)

Prueba si lista es una permutación o no. Devuelve 1 si es afirmativo, y 0 de lo contrario.

0211 Función is_perpendicular

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función is_perpendicular.](#)

1096 Función is_rectangle

ID 1096 · título recuperado de la página española

Sintaxis

is_rectangle(Punto1, Punto2, Punto3, Punto4)

Comprueba si un conjunto de puntos forma un rectángulo o un triángulo rectángulo.

Con cuatro puntos como argumentos, comprueba si los puntos son vértices de un rectángulo.

Devuelve:

0 si no son vértices de un rectángulo;

1 si son vértices de un rectángulo;

2 si son vértices de un cuadrado.

Con tres puntos como argumentos, comprueba si los puntos son vértices de un triángulo rectángulo.

Devuelve:

0 si no son vértices de un triángulo rectángulo;

de lo contrario, el número de posición del vértice común a los dos lados perpendiculares: 1, 2 o 3.

1088 is_rhombus — La prueba verificará si los cuatro puntos de un conjunto son vértices de un rombo o no.

ID 1088 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: is_rhombus

Explicación española: La prueba verificará si los cuatro puntos de un conjunto son vértices de un rombo o no.

Sintaxis

is_rhombus(Punto1, Punto2, Punto3, Punto4)

La prueba verificará si los cuatro puntos de un conjunto son vértices de un rombo o no.

Devuelve

0 si no son

1 si son

2 si son los vértices de un cuadrado

1097 Función is_square

ID 1097 · título recuperado de la página española

Sintaxis

is_square(Punto1, Punto2, Punto3, Punto4)

La prueba verificará si los cuatro puntos de un conjunto son vértices de un cuadrado.

Devuelve 1 si lo son y de lo contrario devuelve 0.

1098 Verifica si el valor es finito

ID 1098 · título recuperado de la página española

Sintaxis

isfinite(Value)

Devuelve false si Value es infinito o un valor indefinido, y true en caso contrario.

1099 Verifica si el valor es infinito

ID 1099 · título recuperado de la página española

Sintaxis

isinf(Value)

Devuelve true si Value es infinito, y false en caso contrario.

0646 Normal de Smith

ID 646 · título recuperado de la página española

Sintaxis

ismith(Matriz_A)

Dado Matriz A, devuelve la forma normal de Smith de una matriz con los coeficientes de Z. Devuelve [U V B] de forma que U y V son invertibles en Z, B es la diagonal, $B[i,i]$ divide $B[i+1,i+1]$ y $B=U*A*V$.

0647 Verifica si el valor está indefinido

ID 647 · título recuperado de la página española

Sintaxis

isnan(Value)

Devuelve true si Value está indefinido (undef), y false en caso contrario.

1092 Función isobarycenter

ID 1092 · título recuperado de la página española

Sintaxis

isobarycenter(Punto1, Punto2, ..., Puntón)

Devuelve el centro hipotético de masa de un conjunto de puntos. Funciona del mismo modo que barycenter, pero asume que todos los puntos tienen el mismo peso.

0159 Polígono regular

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Polígono regular.](#)

0151 Triángulo isósceles

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Triángulo isósceles.](#)

0758 Prueba de primalidad de enteros

ID 758 · título recuperado de la página española

Sintaxis

isprime(Entero)

Devuelve el valor verdadero si el entero es primo; de lo contrario, devuelve falso.

0759 N-ésimo primo

ID 759 · título recuperado de la página española

Sintaxis

`ithprime(Entero)`

Para el entero n , devuelve el n -ésimo número primo, donde n está entre 1 y 200.000.

0840 Funciones en vista CAS: 'J'

ID 840 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'J'

1091 Símbolo de Jacobi

ID 1091 · título recuperado de la página española

Sintaxis

`jacobi_symbol(Entero1, Entero2)`

Devuelve el símbolo de Jacobi de los enteros dados.

0641 Unir una lista de cadenas

ID 641 · título recuperado de la página española

Sintaxis

`join(sep, lst)`
`join(sep, vect)`

Devuelve una cadena que es la concatenación de los elementos en la lista `lst` o el vector `vect`, separados por la cadena proporcionada en el separador `sep`.

0640 Jordan

ID 640 · título recuperado de la página española

Sintaxis

`jordan(Matriz)`

Devuelve la lista hecha por la matriz de paso y la forma de Jordan de una matriz.

0623 Jordan Block

ID 623 · título recuperado de la página española

Sintaxis

`JordanBlock(Expr, n)`

Devuelve una matriz cuadrada $n \times n$ con `Expr` en la diagonal, 1 arriba y 0 en cualquier otro lugar.

0841 Funciones en vista CAS: 'K'

ID 841 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'K'

1103 Núcleo de la matriz

ID 1103 · título recuperado de la página española

Sintaxis

```
ker(Mtrx(M))
```

Devuelve el núcleo de una aplicación lineal de una matriz.

1650 Distribución Kolmogorov-Smirnov

ID 1650 · título recuperado de la página española

Sintaxis

```
kolmogorovd(Real_1 [, ..., Real_N])
```

Devuelve el valor de la distribución de Kolmogorov-Smirnov.

$$1-2*\sum((-1)^{(k-1)}*\exp(-k^2*x^2),k,1,\infty)$$
1651 Prueba de Kolmogorov-Smirnov

ID 1651 · título recuperado de la página española

Sintaxis

```
kolmogorovt(lista1, lista2)
kolmogorovt(lista1, ley_de_distribución)
```

La prueba de igualdad de Kolmogorov-Smirnov a una ley de distribución continua, ya sea entre dos muestras lista1 y lista2 de una distribución desconocida, o entre una muestra lista1 y una ley de distribución específica.

0842 Funciones en vista CAS: 'L'

ID 842 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'L'

0660 Norma L^1

ID 660 · título recuperado de la página española

Sintaxis

```
l1norm(Vector)
```

Devuelve la norma L^1 (suma de los valores absolutos de las coordenadas) de un vector.

0661 Norma L^2

ID 661 · título recuperado de la página española

Sintaxis

```
l2norm(Vector)
```

Devuelve la norma L^2 ($\sqrt{x_1^2+x_2^2+\dots+x_n^2}$) de un vector.

0794 Polinomio de Lagrange

ID 794 · título recuperado de la página española

Sintaxis

```
lagrange([X1, X2, ..., Xn], [Y1, Y2, ..., Yn])
lagrange(Matriz)
```

Dados un vector de abscisas y un vector de ordenadas, devuelve el polinomio de Lagrange para los puntos especificados en los dos vectores.

Esta función también puede tomar una matriz como argumento, con la primera fila con las abscisas y la segunda fila con las ordenadas. Devuelve el polinómico de grado $n-1$ de forma que $P(x_k)=y_k$, para $k=0, 1, \dots, n-1$.

0795 Polinomio de Laguerre

ID 795 · título recuperado de la página española

Sintaxis

laguerre(Entero)

Dado un número entero n , devuelve el polinomio de Laguerre de grado n .

0705 Transformada de Laplace

ID 705 · título recuperado de la página española

Sintaxis

laplace(Expr, [Var], [LapVar])

Devuelve la transformada de Laplace de una expresión.

1104 laplacian — Devuelve el valor laplaciano de una expresión con respecto a un vector de variables.

ID 1104 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: laplacian

Explicación española: Devuelve el valor laplaciano de una expresión con respecto a un vector de variables.

Sintaxis

laplacian(Expr, Vector)

Devuelve el valor laplaciano de una expresión con respecto a un vector de variables.

1105 Generar texto Látex

ID 1105 · título recuperado de la página española

Sintaxis

latex(Expr)

Devuelve la expresión CAS evaluada escrita en formato Látex.

0756 Múltiplo común mínimo

ID 756 · título recuperado de la página española

Sintaxis

lcm(Ent1, Ent2, ...)
lcm(Poli1, Poli2, ...)
lcm(Racional1, Racional2, ...)

Devuelve el múltiplo común mínimo de dos o más polinomios de varias variables, de dos o más números enteros o de dos o más racionales.

1106 lcoeff — Devuelve el coeficiente del término de mayor grado de un polinomio.

ID 1106 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: lcoeff

Explicación española: Devuelve el coeficiente del término de mayor grado de un polinomio.

Sintaxis

lcoeff(Poli)
lcoeff(Vector)
lcoeff(Lista)

Devuelve el coeficiente del término de mayor grado de un polinomio. El polinomio se puede expresar en forma simbólica o como un vector o lista de coeficientes.

1107 ldexp — Para la mantisa `mant` y el exponente `exp`, calcula `mant \* (2^exp)`.

ID 1107 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ldexp

Explicación española: Para la mantisa `mant` y el exponente `exp`, calcula `mant \* (2^exp)`.

Sintaxis

ldexp(mant, exp)

Para la mantisa mant y el exponente exp, calcula $\text{mant} * (2^{\text{exp}})$. Esta operación sirve esencialmente como el inverso de la función frexp().

0683 Lado izquierdo

ID 683 · título recuperado de la página española

Sintaxis

left(Expr1=Expr2)
left(Real1..Real2)
left(Str, n)

Devuelve el lado izquierdo de una ecuación, el extremo izquierdo de un intervalo o los primeros n caracteres de una cadena.

0796 Polinomio de Legendre

ID 796 · título recuperado de la página española

Sintaxis

legendre(Entero)

Dado un número entero n, devuelve el polinomio de Legendre de grado n.

1109 Símbolo de Legendre

ID 1109 · título recuperado de la página española

Sintaxis

legendre_symbol(Entero1, Entero2)

Dados dos enteros, devuelve el símbolo de Legendre del segundo entero, utilizando el polinomio de Legendre cuyo grado es el primer entero.

1110 length — Devuelve la longitud de una lista, una cadena o un conjunto de objetos.

ID 1110 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: length

Explicación española: Devuelve la longitud de una lista, una cadena o un conjunto de objetos.

Sintaxis

length(Lista)
length(Cadena)
length(Objeto)

Devuelve la longitud de una lista, una cadena o un conjunto de objetos.

1111 Divisor común máximo

ID 1111 · título recuperado de la página española

Sintaxis

```
lgcd(Lista)
lgcd(Vector)
lgcd(Entero1, Entero2, ...)
lgcd(Poli1, Poli2, ...)
```

Devuelve el máximo común divisor de un conjunto de números enteros o polinomios contenidos en una lista, un vector, o simplemente ingresados directamente como argumentos.

1112 **lin** — Earización de exponenciales.

ID 1112 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `lin`
Explicación española: Earización de exponenciales.

Sintaxis

```
lin(Expr)
```

Linearización de exponenciales. Devuelve una expresión con los exponenciales linearizados.

0143 **Línea**

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Línea.](#)

1115 **linear_interpolate** — Toma una muestra regular de una línea poligonal definida por una matriz de dos filas.

ID 1115 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `linear_interpolate`
Explicación española: Toma una muestra regular de una línea poligonal definida por una matriz de dos filas.

Sintaxis

```
linear_interpolate(Matriz, Xmin, Xmax, Xstep)
```

Toma una muestra regular de una línea poligonal definida por una matriz de dos filas.

1116 **Regresión lineal**

ID 1116 · título recuperado de la página española

Sintaxis

```
linear_regression(Matriz)
linear_regression(Lista1, Lista2)
```

Dado un conjunto de puntos, devuelve un vector con los coeficientes a y b de $y=a*x+b$, la línea que mejor se ajusta al conjunto de puntos. Los puntos pueden ser los elementos de dos listas o las filas de una matriz.

1117 **Línea horizontal**

ID 1117 · título recuperado de la página española

Sintaxis

```
LineHorz(Exp.)
LineHorz(Real)
```

Se utiliza en la Vista simbólica de la aplicación de Geometría. Dado un número real a o una expresión que evalúa a un número real a , dibuja la línea horizontal $y=a$.

Ejemplos

[Ejemplo gráfico E1] dibuja la línea cuya ecuación es $y=-1$

1168 Función LineTan

ID 1168 · título recuperado de la página española

Sintaxis

LineTan($f(x)$, [Var], Valor)

Dibuja la tangente a $y=f(\text{Var})$ en $\text{Var}=\text{Valor}$.

Ejemplos

[Ejemplo gráfico E1] devuelve $\text{line}(y=x-1)$; es decir, la línea de la tangente de la gráfica de $y=x^2-x$ at $x=1$

1118 Línea vertical

ID 1118 · título recuperado de la página española

Sintaxis

LineVert(Expr)
LineVert(Real)

Se utiliza en la Vista simbólica de la aplicación de Geometría. Dado un número real a o una expresión que evalúa a un número real a , dibuja la línea vertical $x=a$.

Ejemplos

[Ejemplo gráfico E1] dibuja la línea cuya ecuación es $x=2$

0688 Límite

ID 688 · título recuperado de la página española

Sintaxis

limit(Expr, Var, Val, [Dir])

Devuelve el límite (2 lados o 1 lado) de la expresión especificada a medida que la variable dada se acerca a un valor.

El argumento opcional Dir indica el límite de dos lados si 0, un lado desde abajo si -1 y un lado de arriba si 1. Si no se proporciona el cuarto argumento, el límite devuelto es bidireccional.

0718 Solucionador de sistema lineal

ID 718 · título recuperado de la página española

Sintaxis

linsolve([LinEq1, LinEq2, ..., LinEqn], [Var1, Var2, ..., Varn])

Dado un vector de ecuaciones lineales y un vector correspondiente de variables, devuelve la solución para el sistema de ecuaciones lineales.

1119 list2mat — Dada una lista y un entero n, devuelve una matriz de n columnas creada mediante la división de una lista...

ID 1119 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: list2mat

Explicación española: Dada una lista y un entero n , devuelve una matriz de n columnas creada mediante la división de una lista...

Sintaxis

list2mat(Lista, Entero)

Dada una lista y un entero n , devuelve una matriz de n columnas creada mediante la división de una lista en filas, cada una de la cuales consta de n términos. Si el número de elementos de la lista no es divisible por n , la matriz se completa con ceros.

1712 Simplificación LLL

ID 1712 · título recuperado de la página española

Sintaxis

`lll_reduce(Matriz)`

Implementación del algoritmo de reducción con base de retículos de Lenstra-Lenstra-Lovász (LLL). Toma como argumento una matriz invertible con coeficientes enteros.

Devoluciones (S, A, L, O) de modo que:

- las filas de S son una base corta del módulo Z generado por las filas de M
- A es la matriz de cambio de base de la base corta a la base definida por las filas de M ($A \cdot M = S$)
- L es una matriz triangular inferior y el módulo de sus coeficientes no diagonales es menor que $1/2$
- O es una matriz con filas ortogonales de modo que $L \cdot O = S$

1120 Logaritmo natural

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Logaritmo natural.](#)

1121 Lista de nombres de variables

ID 1121 · título recuperado de la página española

Sintaxis

`lname('Expr')`

Devuelve una lista de las variables en una expresión, que debe estar contenida entre comillas simples (').

En CAS, este comando y todas las variables deben estar en minúsculas. En Inicio, el comando y las variables deben estar en mayúsculas.

0720 Recopilar logaritmos

ID 720 · título recuperado de la página española

Sintaxis

`lncollect(Expr)`

Devuelve una expresión reescrita con los logaritmos recopilados. (Aplica $\ln(a) + n \cdot \ln(b) = \ln(a \cdot b^n)$ donde n es un entero.

1122 Ampliar logaritmo

ID 1122 · título recuperado de la página española

Sintaxis

`lnexpand(Expr)`

Devuelve la forma ampliada de una expresión logarítmica.

0171 Locus

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Locus.](#)

1124 Logaritmo general

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Logaritmo general.](#)

1125 Logaritmo común

ID 1125 · título recuperado de la página española

Sintaxis

`log10(Expr)`

Logaritmo común (base 10). Devuelve el logaritmo común de una expresión.

1126 Regresión logarítmica

ID 1126 · título recuperado de la página española

Sintaxis

`logarithmic_regression(Matriz)`
`logarithmic_regression(Lista1, Lista2)`

Dado un conjunto de puntos, devuelve un vector que contiene los coeficientes a y b de $y=a*\ln(x)+b$, la función logarítmica natural que mejor encaja en el conjunto de puntos. Los puntos pueden ser los elementos en dos listas o las filas de una matriz.

1127 **logb** — Dado un número real a y un entero b , Devuelve el logaritmo de a en la base b .

ID 1127 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `logb`

Explicación española: Dado un número real a y un entero b , Devuelve el logaritmo de a en la base b .

Sintaxis

`logb(a, b)`

Dado un número real a y un entero b , Devuelve el logaritmo de a en la base b .

1128 Regresión logística

ID 1128 · título recuperado de la página española

Sintaxis

`logistic_regression(Lst(L), Real(x0), Real(y0))`

Devuelve $[y, y', C, y'_{\max}, x_{\max}, R]$ donde y es una función de logística (solución de $y'/y=a*y+b$), de forma que $y(x_0)=y_0$ y donde $[y'(x_0), y'(x_0+1)\dots]$ es la mejor aproximación de L .

0713 Descomposición LU

ID 713 · título recuperado de la página española

Sintaxis

`lu(A)`

Factoriza una matriz cuadrada A en un vector de permutación p y dos matrices L y U , donde p representa la permutación de filas, L es una matriz triangular inferior con unos en su diagonal, y U es una matriz triangular superior. Si P es la matriz de permutación correspondiente a p , entonces $P*A=L*U$.

1130 Lista de variables y expresiones

ID 1130 · título recuperado de la página española

Sintaxis

`lvar(Expr)`

Dada una expresión, devuelve una lista de las funciones de la expresión que utiliza las variables, incluyendo la ocurrencia de las mismas variables.

0843 Funciones en vista CAS: 'M'

Lista de funciones y comandos en la vista CAS que comienzan con 'M'

0603 Crear lista

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Crear lista.](#)

0620 Crear matriz

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Crear matriz.](#)

0535 Mantisa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Mantisa.](#)

1131 Función map

Sintaxis

```
map(Matrix, Var→Function)
map(Matrix, Var→Test)
```

Hay dos formas de esta función. El segundo argumento siempre es una asignación de una variable a una expresión.

Ejemplos

[Ejemplo gráfico E1] Si la expresión es una función de la variable, se aplica a cada elemento del vector o matriz pasado como primer argumento, y se devuelve el vector o matriz resultante.
[Ejemplo gráfico E2] Si la expresión es una prueba booleana, se prueba cada elemento del vector o matriz, y los resultados se devuelven como un vector o matriz de valores 0 y 1. Cada prueba devuelve 0 si falla o 1 si pasa.

1132 mat2list — Devuelve una lista que contiene los elementos de la matriz dada.

Referencia traducida

Identificador técnico: mat2list

Explicación española: Devuelve una lista que contiene los elementos de la matriz dada.

Sintaxis

```
mat2list(Matriz)
```

Devuelve una lista que contiene los elementos de la matriz dada.

1133 matpow — Calcula la potencia de n de una matriz mediante el uso de la forma normal de Jordan.

Referencia traducida

Identificador técnico: matpow

Explicación española: Calcula la potencia de n de una matriz mediante el uso de la forma normal de Jordan.

Sintaxis

```
matpow(Matriz, Ent(n))
```

Calcula la potencia de n de una matriz mediante el uso de la forma normal de Jordan.

1134 matrix — Devuelve una matriz con `rows` filas y `columns` columnas.

ID 1134 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `matrix`

Explicación española: Devuelve una matriz con ``rows`` filas y ``columns`` columnas.

Sintaxis

```
matrix(rows, columns, func)
```

Devuelve una matriz con `rows` filas y `columns` columnas. Cada elemento se obtiene evaluando la función `func` con dos argumentos. El primer argumento representa el índice de fila y el segundo, el índice de columna.

0538 Máximo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Máximo.](#)

0659 Norma máx.

ID 659 · título recuperado de la página española

Sintaxis

```
maxnorm(Vector)  
maxnorm(Matriz)
```

Devuelve la norma l_∞ (lo máximo de los valores absolutos de las coordenadas) de un vector o matriz.

1135 Real máximo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Real máximo.](#)

1136 Promedio

ID 1136 · título recuperado de la página española

Sintaxis

```
mean(Lista1, [Lista2])  
mean(Matriz)
```

Devuelve el promedio aritmético de una lista (con una lista opcional en forma de lista de factores de ponderación). Con una matriz como argumento, devuelve el promedio de las columnas.

1137 Mediana

ID 1137 · título recuperado de la página española

Sintaxis

```
median(Lista1, [Lista2])  
median(Matriz)
```

Devuelve la mediana de una lista o vector (con una lista opcional como una lista de pesos). Con una matriz como argumento, devuelve la mediana de las columnas.

0147 Mediana

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Mediana.](#)

1139 Miembro

ID 1139 · título recuperado de la página española

Sintaxis

```
member(Elemento, Lista)  
member(Elemento, Vector)
```

Dada una lista o vector y un elemento, devuelve el índice de la primera ocurrencia del elemento en la lista o vector; si el elemento no aparece en la lista o vector, devuelve 0. Similar a `contains`, excepto que el elemento está primero en el orden del argumento.

1140 Medio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Medio](#).

0133 Punto medio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Punto medio](#).

0539 Mínimo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Mínimo](#).

1141 Real mínimo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Real mínimo](#).

1142 Diferencia de dos listas o vectores

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Diferencia de dos listas o vectores](#).

0625 Isometría

ID 625 · título recuperado de la página española

Sintaxis

```
mkisom(Vect, Sign)
```

Devuelve la matriz de una isometría dada por sus elementos propios. Signo puede ser +1 para una isometría directa o -1 para una isometría indirecta.

Ejemplos

```
[Ejemplo gráfico E1] (en modo radianes)  
[Ejemplo gráfico E2]
```

1143 Convertir al sistema métrico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Convertir al sistema métrico](#).

1144 modgcd — Utiliza el algoritmo modular para devolver el máximo común divisor de dos polinomios.

ID 1144 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `modgcd`

Explicación española: Utiliza el algoritmo modular para devolver el máximo común divisor de dos polinomios.

Sintaxis

```
modgcd(Poli1, Poli2)
```

Utiliza el algoritmo modular para devolver el máximo común divisor de dos polinomios.

1146 Multiplicar fila

ID 1146 · título recuperado de la página española

Sintaxis

mRow(Expr, Matriz, Entero)

Dada una expresión, una matriz y un número entero n, multiplica la fila n de la matriz por la expresión.

1148 mult_c_conjugate — Si la expresión dada tiene un denominador complejo, devuelve la expresión después de que tanto el...

ID 1148 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: mult_c_conjugate

Explicación española: Si la expresión dada tiene un denominador complejo, devuelve la expresión después de que tanto el...

Sintaxis

mult_c_conjugate(Expr)

Si la expresión dada tiene un denominador complejo, devuelve la expresión después de que tanto el numerador como el denominador hayan sido multiplicados por el conjugado complejo del denominador. Si la expresión compleja dada no tiene un denominador complejo, devuelve la expresión luego de que tanto el numerador como el denominador hayan sido multiplicados por el conjugado complejo del numerador.

1149 mult_conjugate — Toma una expresión en la que el numerador o el denominador contienen una raíz cuadrada.

ID 1149 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: mult_conjugate

Explicación española: Toma una expresión en la que el numerador o el denominador contienen una raíz cuadrada.

Sintaxis

mult_conjugate(Expr)

Toma una expresión en la que el numerador o el denominador contienen una raíz cuadrada. Si el denominador contiene una raíz cuadrada, devuelve la expresión una vez multiplicados el numerador y el denominador por el complejo conjugado del denominador. Si el denominador no contiene una raíz cuadrada, devuelve la expresión una vez multiplicados el numerador y el denominador por el complejo conjugado del numerador.

1653 Distribución Multinomial

ID 1653 · título recuperado de la página española

Sintaxis

multinomial(Enteron, VectorP, VectorK)

Calcula la probabilidad de éxito de VectorK de Enteron intentos, cada uno con una probabilidad de éxito de VectorP.

La distribución multinomial es una forma más general de la Distribución Binomial donde cada resultado puede tener dos o más resultados posibles.

0844 Funciones en vista CAS: 'N'

ID 844 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'N'

1150 Derivado numérico

ID 1150 · título recuperado de la página española

Sintaxis

```
nDeriv(Expr, Var)
nDeriv(Expr, Var, Step)
```

Devuelve una aproximación de la derivada de Expr con respecto a Var.

nDeriv(Expr, Var) usa el tamaño de paso predeterminado 0.001.

En nDeriv(Expr, Var, Step), el tercer argumento Step establece el tamaño de paso.

Si Step es un número real, el comando usa ese valor como tamaño de paso.

Si Step es una variable, el comando devuelve la expresión de diferencia central usando esa variable como tamaño de paso.

El resultado se basa en la fórmula de diferencia central:

$(f(\text{Var}+h) - f(\text{Var}-h)) / (2 \cdot h)$

1151 Binomial negativa discreta

ID 1151 · título recuperado de la página española

Sintaxis

```
negbinomial(n, k, x)
```

Función de densidad de probabilidad binomial negativa

Calcula la densidad de probabilidad de la distribución binomial negativa en x, con los parámetros n y k.

1152 Binomial negativa acumulada

ID 1152 · título recuperado de la página española

Sintaxis

```
negbinomial_cdf(n, k, x, [x2])
```

Función de distribución binomial negativa acumulada

Devuelve la probabilidad de cola inferior de la función de densidad de probabilidad binomial negativa para el valor x, con los parámetros n y k.

1153 Binomial negativa acumulada inversa

ID 1153 · título recuperado de la página española

Sintaxis

```
negbinomial_icdf(n, k, p)
```

Función de distribución binomial negativa acumulada inversa

Devuelve el valor de x de forma tal que la probabilidad de cola inferior binomial negativa de x, con los parámetros n y k, sea p.

1154 Método de Newton

ID 1154 · título recuperado de la página española

Sintaxis

```
newton(Expr, Var, [Supuesto], [Entero])
```

Usa el método de Newton para calcular la raíz de una función, empezando por Supuesto y calculando Entero iteraciones. De forma predeterminada, Entero es 20.

0760 Siguiendo primo

ID 760 · título recuperado de la página española

Sintaxis

`nextprime(Entero)`

Devuelve el menor número primo, mayor que el argumento.

1692 Sin operación

ID 1692 · título recuperado de la página española

Sintaxis

`nop()`

La función CAS sin operación. En la evaluación, no se producirá ninguna operación.

Esta función puede ser útil para algunos casos de uso avanzados en la programación de la función CAS.

1156 normal — Devuelve la forma irreducible expandida de una expresión.

ID 1156 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `normal`

Explicación española: Devuelve la forma irreducible expandida de una expresión.

Sintaxis

`normal(Expr)`

Devuelve la forma irreducible expandida de una expresión.

0578 Densidad normal

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Densidad normal.](#)

0586 Normal acumulada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Normal acumulada.](#)

0594 Normal acumulada inversa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Normal acumulada inversa.](#)

1157 Normalizar

ID 1157 · título recuperado de la página española

Sintaxis

`normalize(Vector)`
`normalize(Compleja)`

Dado un vector, lo devuelve dividido por su norma l2 (donde norma l2 es la raíz cuadrada de las coordenadas del vector).

Dado un número complejo, lo devuelve dividido por su módulo.

Es también una opción para campo de gráfico. En este caso, el término aparece último en el conjunto de argumentos y el resultado es que los segmentos del campo de direcciones reciben igual longitud.

0682 Numerador simplificado

ID 682 · título recuperado de la página española

Sintaxis

```
numer(a/b)
```

Para los enteros a y b, devuelve el numerador de la fracción a/b tras la simplificación.

0845 Funciones en vista CAS: 'O'

ID 845 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'O'

1161 Prueba de imparidad

ID 1161 · título recuperado de la página española

Sintaxis

```
odd(Entero)
```

Devuelve 1 si el número entero dado es impar, de lo contrario, devuelve 0.

0717 ODE Solver

ID 717 · título recuperado de la página española

Sintaxis

```
odesolve(Expr, VectVar, InicVect, ValFinal, [tstep=Val, Curva])
```

Solucionador de ecuación diferencial ordinaria

Resuelve una ecuación diferencial ordinaria dada por Expr, con las variables declaradas en VectVar y las condiciones iniciales de las variables declaradas en InicVect. Por ejemplo, `odesolve(f(t,y),[t,y],[t0,y0],t1)` devuelve la solución aproximada de $y'=f(t,y)$ para las variables t e y con condiciones iniciales $t=t_0$ e $y=y_0$.

1162 op — Devuelve los argumentos de un operador como una secuencia.

ID 1162 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: op

Explicación española: Devuelve los argumentos de un operador como una secuencia.

Sintaxis

```
op(expression)
```

Devuelve los argumentos de un operador como una secuencia.

1163 open_polygon — Conecta un conjunto de puntos con segmentos de línea, en el orden dado, para producir un polígono.

ID 1163 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: open_polygon

Explicación española: Conecta un conjunto de puntos con segmentos de línea, en el orden dado, para producir un polígono.

Sintaxis

```
open_polygon(punto1, punto2, ..., punto1)
open_polygon(punto1, punto2, ..., punton)
```

Conecta un conjunto de puntos con segmentos de línea, en el orden dado, para producir un polígono. Si el último punto es el mismo que el primero, el polígono es cerrado; de lo contrario, es abierto.

Ejemplos

[Ejemplo gráfico E1] dibuja un triángulo rectángulo

1165 ord — Para una cadena, devuelve el código ASCII del primer carácter de `String`.

ID 1165 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `ord`

Explicación española: Para una cadena, devuelve el código ASCII del primer carácter de `String`.

Sintaxis

```
ord(String)
ord(List)
ord(Vector)
```

Para una cadena, devuelve el código ASCII del primer carácter de `String`.

Para una lista o vector de cadenas, devuelve una lista de códigos ASCII de los primeros caracteres de los elementos de `List` o `Vector`.

1166 order_size — Devuelve el resto (término O) de una expansión de serie.: $\text{limit}(x^a \cdot \text{order_size}(x), x=0)=0$ si $a>0$

ID 1166 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `order_size`

Explicación española: Devuelve el resto (término O) de una expansión de serie.: $\text{limit}(x^a \cdot \text{order_size}(x), x=0)=0$ si $a>0$

Sintaxis

```
order_size(Expr)
```

Devuelve el resto (término O) de una expansión de serie.:

$\text{limit}(x^a \cdot \text{order_size}(x), x=0)=0$ si $a>0$

0194 Ordenada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ordenada.](#)

1167 Función orthocenter

ID 1167 · título recuperado de la página española

Sintaxis

```
orthocenter(Triángulo)
orthocenter(Punto1, Punto2, Punto3)
```

Devuelve el ortocentro de un triángulo; es decir, la intersección de las tres altitudes de un triángulo. El argumento puede ser el nombre de un triángulo o tres puntos no colineales que definen un triángulo. En el último caso, no es necesario dibujar el triángulo.

1160 orthogonal — $(A, \text{line}(B,C))$ dibuja el plano ortogonal de la línea BC a través de A.

ID 1160 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `orthogonal`

Explicación española: $(A, \text{line}(B,C))$ dibuja el plano ortogonal de la línea BC a través de A.

Sintaxis

```
orthogonal(Punto, Línea)
orthogonal(Punto, Plano)
```

`orthogonal(A, line(B,C))` dibuja el plano ortogonal de la línea BC a través de A.

0846 Funciones en vista CAS: 'P'

Lista de funciones y comandos en la vista CAS que comienzan con 'P'

1701 p1oc2 — Devuelve el producto de permutación de permutación y ciclo.

ID 1701 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: p1oc2

Explicación española: Devuelve el producto de permutación de permutación y ciclo.

Sintaxis

p1oc2(permutación, ciclo)

Devuelve el producto de permutación de permutación y ciclo.

1700 p1op2 — Devuelve el producto de permutación de permutación1 y permutación2.

ID 1700 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: p1op2

Explicación española: Devuelve el producto de permutación de permutación1 y permutación2.

Sintaxis

p1op2(permutación1, permutación2)

Devuelve el producto de permutación de permutación1 y permutación2.

1169 pa2b2 — Toma un número entero primo n congruente a 1 módulo 4 y devuelve [a,b], de forma que $a^2+b^2=n$.

ID 1169 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: pa2b2

Explicación española: Toma un número entero primo n congruente a 1 módulo 4 y devuelve [a,b], de forma que $a^2+b^2=n$.

Sintaxis

pa2b2(Entero)

Toma un número entero primo n congruente a 1 módulo 4 y devuelve [a,b], de forma que $a^2+b^2=n$.

1170 Aproximación de Padé

ID 1170 · título recuperado de la página española

Sintaxis

pade(Expr, Var, Enteron, Enterop)

Devuelve la aproximación de Padé de una expresión, es decir, una fracción racional P/Q , de tal manera que $P/Q \equiv \text{Expr} \pmod{x^{n+1}}$ o $\pmod{n}$ con $\text{grado} < p$.

0169 Parábola

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Parábola.](#)

0144 Paralelo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Paralelo.](#)

0155 Paralelogramo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Paralelogramo.](#)

0197 Paramétrica

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Paramétrica.](#)

1171 Parte de expresión

ID 1171 · título recuperado de la página española

Sintaxis

`part(Expr, Entero)`

Devuelve la n -ésima subexpresión de una expresión.

0678 Descomposición de fracción parcial

ID 678 · título recuperado de la página española

Sintaxis

`partfrac(RatFrac)`

Realiza una descomposición en fracciones parciales en una fracción.

0777 Raíces a coeficientes

ID 777 · título recuperado de la página española

Sintaxis

`pcoeff(Vector)`
`pcoeff(Lista)`

Dada una lista o vector que contiene las raíces de un polinomio, devuelve un vector que contiene (en orden decreciente) los coeficientes del polinomio univariable que poseen esas raíces.

0201 Perímetro

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Perímetro.](#)

1232 perimeterat

ID 1232 · título recuperado de la página española

Sintaxis

`perimeterat(polygon, point)`
`perimeterat(circle, point)`

Se utiliza en la Vista simbólica de la aplicación Geometría. Muestra el perímetro de un polígono o la circunferencia de un círculo. La medida se muestra, con una etiqueta, en el punto dado de la Vista de gráfico.

Ejemplos

`perimeterat(circle(x^2+y^2=1), point(-4,4))`
muestra "`pcircle(x^2+y^2=1)= 2*π`" en el punto `(-4, 4)`

0570 Permutaciones

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Permutaciones.](#)

1702 Permutación inversa

ID 1702 · título recuperado de la página española

Sintaxis

`perminv(permutación)`

Devuelve la permutación inversa de permutación.

1694 permu2cycles — Convierte permutación a un producto de ciclos disociados.

ID 1694 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `permu2cycles`

Explicación española: Convierte permutación a un producto de ciclos disociados.

Sintaxis

`permu2cycles(permutación)`

Convierte permutación a un producto de ciclos disociados.

1696 permu2mat — Devuelve la matriz donde las filas de la matriz de identidad se cambian con permutación.

ID 1696 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `permu2mat`

Explicación española: Devuelve la matriz donde las filas de la matriz de identidad se cambian con permutación.

Sintaxis

`permu2mat(permutación)`

Devuelve la matriz donde las filas de la matriz de identidad se cambian con permutación.

1703 Orden de permutación

ID 1703 · título recuperado de la página española

Sintaxis

`permuorder(permutación)`

Devuelve el orden de permutación.

1172 Bisector perpendicular

ID 1172 · título recuperado de la página española

Sintaxis

`perpen_bisector(Segmento)`
`perpen_bisector(Punto1, Punto2)`

Dibuja el bisector perpendicular de un segmento. El segmento se define por su nombre o sus dos extremos.

Ejemplos

[Ejemplo gráfico E1] dibuja el bisector perpendicular de un segmento C.

[Ejemplo gráfico E2] dibuja el bisector perpendicular de un segmento AB.

[Ejemplo gráfico E3] dibuja el bisector perpendicular de un segmento cuyos extremos tienen las coordenadas (3, 2) y (0, 1); es decir, la línea cuya ecuación es $y=x/3+1$.

0145 Perpendicular

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Perpendicular.](#)

1173 Función piecewise

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función piecewise.](#)

0635 Pivote

ID 635 · título recuperado de la página española

Sintaxis

`pivot(matriz, n, m)`

Dada una matriz, un número de fila n , y un número de columna m , utiliza la eliminación de Gauss para devolver una matriz con ceros en la columna m , excepto en el caso de que el elemento de la columna m y la fila n se mantenga como pivote.

0801 Trazar contorno

ID 801 · título recuperado de la página española

Sintaxis

`plotcontour(Expr [, Vars [, Values]])`

Dibuja las líneas de contorno de la superficie $z = \text{Expr}$.

`Vars` especifica una lista o un vector de dos variables. La primera variable se usa para el eje horizontal y la segunda para el eje vertical. Si no se especifica `Vars`, se usan las variables x e y . `Expr` puede depender de una o de ambas variables.

`Values` especifica una lista o un vector no vacío de niveles de contorno numéricos. Si se especifica `Values`, el comando dibuja las líneas de contorno $z = z_1, \dots, z = z_n$ en los niveles especificados por `Values`. Si no se especifica `Values`, los niveles de contorno se eligen automáticamente.

Se puede usar en la Vista de gráfico o en la Vista simbólica de la aplicación Geometría, así como en la Vista CAS.

Ejemplos

[Ejemplo gráfico E1] dibuja tres líneas de contorno para la expresión dada.

0179 Trazar campo de direcciones

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar campo de direcciones.](#)

0173 Trazar función

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar función.](#)

0177 Trazar implícito

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar implícito.](#)

0181 Trazar lista

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar lista.](#)

0180 Trazar EDO

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar EDO.](#)

0178 Trazar inecuación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar inecuación.](#)

0174 Trazar paramétrico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar paramétrico.](#)

0175 Trazar Polar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar Polar.](#)

0176 Trazar secuencia

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar secuencia.](#)

0780 Polinomio mínimo

ID 780 · título recuperado de la página española

Sintaxis

`pmin(Matriz, [Var])`

Con solo una matriz como argumento, devuelve el polinomio mínimo de x de una matriz escrita como una lista de sus coeficientes. Con una matriz y una variable como argumentos, devuelve el polinomio mínimo de la matriz escrita en forma simbólica con respecto a la variable.

0131 Punto

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Punto.](#)

1174 Función point2d

ID 1174 · título recuperado de la página española

Sintaxis

`point2d(punto1, punto2, ..., punton)`

Vuelve a distribuir de manera aleatoria un conjunto de puntos de modo que, para cada punto, x está en el intervalo $[-5, 5]$ e y está en el intervalo $[-5, 5]$. Cualquier movimiento posterior de uno de los puntos volverá a distribuir aleatoriamente todos los puntos.

0584 Densidad de Poisson

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Densidad de Poisson.](#)

0592 Poisson acumulada

ID 592 · título recuperado de la página española

Sintaxis

`POISSON_CDF( $\mu$ ,  $k$  [,  $k2$ ])`

Función de distribución de Poisson acumulada

Devuelve la probabilidad de que ocurran k o menos incidencias de un evento en un intervalo de tiempo, con μ incidencias previstas (o promedio). Con el tercer argumento opcional $k2$, devuelve la probabilidad entre K y $K2$ incidencias.

0601 Inversa de Poisson acumulada

ID 601 · título recuperado de la página española

Sintaxis

`POISSON_ICDF( $\mu$ , p)`

Inversa de la función de distribución de Poisson acumulada.

Devuelve el valor K de tal manera que la probabilidad de que ocurran K o menos incidencias de un evento en un intervalo de tiempo, con μ incidencias previstas (o promedio) del evento en el intervalo, es p.

1260 polar — Devuelve la línea polar del punto dado como polo con respecto al círculo dado.

ID 1260 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `polar`

Explicación española: Devuelve la línea polar del punto dado como polo con respecto al círculo dado.

Sintaxis

`polar(Círculo, Punto)`
`polar(Círculo, Compleja)`

Devuelve la línea polar del punto dado como polo con respecto al círculo dado.

0195 Coordenadas polares

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Coordenadas polares.](#)

1272 Punto polar

ID 1272 · título recuperado de la página española

Sintaxis

`polar_point(Radio, Ángulo)`

Dados el radio y el ángulo de un punto en forma polar, devuelve el punto con las coordenadas rectangulares en forma compleja.

1261 pole — Devuelve el polo de la línea dada con respecto al círculo dado.

ID 1261 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `pole`

Explicación española: Devuelve el polo de la línea dada con respecto al círculo dado.

Sintaxis

`pole(Círculo, Línea)`

Devuelve el polo de la línea dada con respecto al círculo dado.

0776 Coeficientes a polinomio

ID 776 · título recuperado de la página española

Sintaxis

`poly2symb(Vector [, Var])`

Con un vector como argumento, devuelve un polinomio en x con coeficientes (en orden decreciente) que se obtienen a partir del vector del argumento. Con una variable como segundo argumento, devuelve un polinomio similar en esa variable.

1224 Evaluación de polinomios

ID 1224 · título recuperado de la página española

Sintaxis

`polyEval(Vector, Real)`

Dado un polinomio definido por un vector de coeficientes y un valor real n , evalúa el polinomio en ese valor.

1245 forma polinómica

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de forma polinómica.](#)

0158 Polígono

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Polígono.](#)

1253 polygonplot — Se utiliza en la Vista simbólica de la aplicación de Geometría.

ID 1253 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `polygonplot`

Explicación española: Se utiliza en la Vista simbólica de la aplicación de Geometría.

Sintaxis

`polygonplot(Mtrx)`

Se utiliza en la Vista simbólica de la aplicación de Geometría. Dada una matriz $n \times m$, dibuja y conecta los puntos (x_k, y_k) , donde x_k es el elemento en la fila k y columna 1 e y_k es el elemento en la fila k y la columna j (con j fijo para $k=1$ a las filas n). Por lo tanto, cada emparejamiento de columna genera su propia figura, resultando en figuras $m-1$.

Ejemplos

[Ejemplo gráfico E1] dibuja dos figuras, cada una con tres puntos conectados por segmentos.

1252 polygonscatterplot — Se utiliza en la Vista simbólica de la aplicación de Geometría.

ID 1252 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `polygonscatterplot`

Explicación española: Se utiliza en la Vista simbólica de la aplicación de Geometría.

Sintaxis

`polygonscatterplot(Matriz)`

Se utiliza en la Vista simbólica de la aplicación de Geometría. Dada una matriz $n \times m$, dibuja y conecta los puntos (x_k, y_k) , donde x_k es el elemento en la fila k y columna 1 e y_k es el elemento en la fila k y la columna j (con j fijo para $k=1$ a las filas n). Por lo tanto, cada emparejamiento de columna genera su propia figura, resultando en figuras $m-1$.

Ejemplos

[Ejemplo gráfico E1] dibuja dos figuras, cada una con tres puntos conectados por segmentos.

1244 Regresión polinómica

ID 1244 · título recuperado de la página española

Sintaxis

```
polynomial_regression(Lista1, Lista2, Entero)  
polynomial_regression(Matriz, Entero)
```

Dado un conjunto de puntos definidos por dos listas o una matriz y un número entero positivo n, devuelve un vector que contiene los coeficientes del polinomio de enésima orden que mejor aproxima a los puntos dados.

1225 pop — Devuelve el último elemento de la lista o vector, eliminándolo simultáneamente de la lista o vector.

ID 1225 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: pop

Explicación española: Devuelve el último elemento de la lista o vector, eliminándolo simultáneamente de la lista o vector.

Sintaxis

```
pop(List)  
pop(Vector)
```

Devuelve el último elemento de la lista o vector, eliminándolo simultáneamente de la lista o vector.

1226 potential — Devuelve una función cuyo gradiente es el campo vectorial definido por un vector y un vector de...

ID 1226 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: potential

Explicación española: Devuelve una función cuyo gradiente es el campo vectorial definido por un vector y un vector de...

Sintaxis

```
potential(Vector1, Vector2)
```

Devuelve una función cuyo gradiente es el campo vectorial definido por un vector y un vector de variables.

0725 pow2exp — Devuelve una expresión con potencias reescritas como un exponencial.

ID 725 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: pow2exp

Explicación española: Devuelve una expresión con potencias reescritas como un exponencial.

Sintaxis

```
pow2exp(Expr)
```

Devuelve una expresión con potencias reescritas como un exponencial. Esencialmente el inverso de exp2pow.

0721 Ampliar potencia

ID 721 · título recuperado de la página española

Sintaxis

```
powexpand(Expr)
```

Vuelve a escribir una expresión que contiene una potencia que es una suma o un producto como un producto de potencias. Aplica $a^{(b+c)} = (a^b) \cdot (a^c)$.

1243 Regresión de potencia

ID 1243 · título recuperado de la página española

Sintaxis

```
power_regression(List1, List2)
power_regression(Vector1, Vector2)
power_regression(Matrix)
```

Dado un conjunto de puntos definidos por dos listas, dos vectores o una matriz, devuelve un vector que contiene los coeficientes m y b de $y=b*x^m$, la función potencia que mejor aproxima los puntos dados.

1266 powerpc — Dados un círculo y un punto, devuelve la diferencia entre el cuadrado de la distancia desde el punto...

ID 1266 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: powerpc

Explicación española: Dados un círculo y un punto, devuelve la diferencia entre el cuadrado de la distancia desde el punto...

Sintaxis

```
powerpc(Círculo, Punto)
```

Dados un círculo y un punto, devuelve la diferencia entre el cuadrado de la distancia desde el punto hasta el centro del círculo y el cuadrado del radio del círculo.

0766 Módulo y potencia de entero

ID 766 · título recuperado de la página española

Sintaxis

```
powmod(a, n, p [, Expr, Var])
```

Para los números enteros a , n , y p , devuelve $a^n \bmod p$.

1227 Prepend

ID 1227 · título recuperado de la página española

Sintaxis

```
prepend(Lista, Elemento)
prepend(Vector, Elemento)
```

Agrega un elemento al comienzo de una lista o vector.

0699 preval — Devuelve $F(b) - F(a)$.

ID 699 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: preval

Explicación española: Devuelve $F(b) - F(a)$.

Sintaxis

```
preval(F(var), Real(a), Real(b), [Var])
```

Devuelve $F(b) - F(a)$.

0761 Primo anterior

ID 761 · título recuperado de la página española

Sintaxis

```
prevprime(Entero)
```

Devuelve el mayor número primo, menor que el argumento.

1183 primpart — Devuelve un polinomio dividido por el máximo común divisor de sus coeficientes.

ID 1183 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: primpart

Explicación española: Devuelve un polinomio dividido por el máximo común divisor de sus coeficientes.

Sintaxis

primpart(Poli, [Var])

Devuelve un polinomio dividido por el máximo común divisor de sus coeficientes.

1257 product — Con una expresión como primer argumento, devuelve el producto de soluciones cuando la variable de la...

ID 1257 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: product

Explicación española: Con una expresión como primer argumento, devuelve el producto de soluciones cuando la variable de la...

Sintaxis

product(Expr, [Var], [Mín.], [Máx.], [Incrm.])
product(Lista)
product(Matriz)

Con una expresión como primer argumento, devuelve el producto de soluciones cuando la variable de la expresión va de un valor mínimo a un valor máximo por incremento dado. Si no se proporciona un incremento, se entiende como 1.

Con una lista como primer argumento, devuelve el producto de los valores de la lista.

Con una matriz como primer argumento, devuelve el producto de la matriz elemento por elemento.

0188 Proyección

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Proyección.](#)

0769 Raíces de polinomios

ID 769 · título recuperado de la página española

Sintaxis

proot(Poli)
proot(Vector)

Devuelve todas las raíces calculadas de un polinomio dado por sus coeficientes (puede no funcionar si las raíces no son simples).

1197 Fracción propia

ID 1197 · título recuperado de la página española

Sintaxis

propfrac(Fracción)
propfrac(RatFrac)

Devuelve una fracción o fracción racional A/B simplificada a $Q+R/B$, con $R < B$ (o con el grado de R menor que el grado de B).

0667 Psi — Devuelve el valor de la derivada nth de la función digamma en $x=a$, donde la función digamma es la...

ID 667 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Psi

Explicación española: Devuelve el valor de la derivada nth de la función digamma en $x=a$, donde la función digamma es la...

Sintaxis

`Psi(Real(a), Ent(n))`

Devuelve el valor de la derivada nth de la función digamma en $x=a$, donde la función digamma es la primera derivada de $\ln(\Gamma(x))$.

1228 ptayl — Dadoun polinomio P y un valor a, devuelve la Q del polinomio de Taylor, de tal manera que $P(x)=Q(x-a)$.

ID 1228 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ptayl

Explicación española: Dadoun polinomio P y un valor a, devuelve la Q del polinomio de Taylor, de tal manera que $P(x)=Q(x-a)$.

Sintaxis

`ptayl(Poli, Valor, [Var])`

Dadoun polinomio P y un valor a, devuelve la Q del polinomio de Taylor, de tal manera que $P(x)=Q(x-a)$.

1190 Purgar variable

ID 1190 · título recuperado de la página española

Sintaxis

`purge(Var)`

Elimina la asignación de un nombre de variable en la Vista CAS. Por ejemplo, si se define f, purgar(f) elimina esa definición y devuelve f a su estado simbólico.

1321 Compatibilidad de CAS Python

ID 1321 · título recuperado de la página española

Sintaxis

`python_compat([Entero])`

Establece la compatibilidad con la sintaxis de Python CAS (1) o no (0). Sin argumento, devuelve el modo de compatibilidad actual.

0847 Funciones en vista CAS: 'Q'

ID 847 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'Q'

1222 q2a — Dada una forma cuadrática y un vector de variables, devuelve la matriz simétrica de la forma cuadrática...

ID 1222 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: q2a

Explicación española: Dada una forma cuadrática y un vector de variables, devuelve la matriz simétrica de la forma cuadrática...

Sintaxis

`q2a(QuadExpr, Vector)`

Dada una forma cuadrática y un vector de variables, devuelve la matriz simétrica de la forma cuadrática con respecto a las variables dadas.

0652 Factorización QR

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Factorización QR.](#)

0153 Cuadrilátero

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Cuadrilátero.](#)

1240 quantile — Dada una lista o un vector y un valor de cuantil entre 0 y 1, devuelve el cuantil correspondiente de los...

ID 1240 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: quantile

Explicación española: Dada una lista o un vector y un valor de cuantil entre 0 y 1, devuelve el cuantil correspondiente de los...

Sintaxis

```
quantile(Lista, Valor)
quantile(Vector, Valor)
```

Dada una lista o un vector y un valor de cuantil entre 0 y 1, devuelve el cuantil correspondiente de los elementos de la lista o del vector.

1238 quartile1 — Dados una lista o vector, devuelve el primer cuartil de los elementos de la lista o el vector.

ID 1238 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: quartile1

Explicación española: Dados una lista o vector, devuelve el primer cuartil de los elementos de la lista o el vector.

Sintaxis

```
quartile1(Lista)
quartile1(Vector)
quartile1(Matriz)
```

Dados una lista o vector, devuelve el primer cuartil de los elementos de la lista o el vector. Dada una matriz, devuelve el primer cuartil de las columnas de la matriz.

1239 quartile3 — Dada una lista o vector, devuelve el tercer cuantil de los elementos de la lista o vector.

ID 1239 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: quartile3

Explicación española: Dada una lista o vector, devuelve el tercer cuantil de los elementos de la lista o vector.

Sintaxis

```
quartile3(Lista)
quartile3(Vector)
quartile3(Matriz)
```

Dada una lista o vector, devuelve el tercer cuantil de los elementos de la lista o vector. Dada una matriz, devuelve el tercer cuantil de las columnas de la matriz.

1237 Cuantiles

ID 1237 · título recuperado de la página española

Sintaxis

```
quartiles(Lista)
quartiles(Vector)
quartiles(Matriz)
```

Devuelve una matriz que contiene el mínimo, primer cuantil, mediana, tercer cuantil y máximo de los elementos de una lista o vector. Con una matriz como argumento, devuelve el Resumen de 5-número de las columnas de la matriz.

0782 Cociente

ID 782 · título recuperado de la página española

Sintaxis

```
quo(Lista1, Lista2, [Var])
quo(Poli1, Poli2, [Var])
```

Devuelve un vector que contiene los coeficientes del cociente euclidiano de dos polinomios. Los polinomios pueden escribirse como una lista de coeficientes o en forma simbólica.

1181 Cociente y resto

ID 1181 · título recuperado de la página española

Sintaxis

```
quorem(Poli1, Poli2)
quorem(Vector1, Vector2)
```

Devuelve el cociente y el resto Euclidiano del cociente de 2 polinomios, cada uno se expresa en forma simbólica directamente o como un vector de coeficientes. Si los polinomios se expresan como vectores de sus coeficientes, este comando devuelve un vector similar del cociente y un vector del resto.

1223 Argumento de cotización

ID 1223 · título recuperado de la página española

Sintaxis

```
quote(Expr)
```

Devuelve el argumento no evaluado.

Ejemplos

```
[Ejemplo gráfico E1]
[Ejemplo gráfico E2]
```

También puede utilizar este comando para eliminar una variable.

`a:=quote(a)` elimina la variable `a`

0848 Funciones en vista CAS: 'R'

ID 848 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'R'

1262 radical_axis — Devuelve la línea cuyos puntos tienen los mismos valores de `powerpc` para los dos círculos dados.

ID 1262 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `radical_axis`

Explicación española: Devuelve la línea cuyos puntos tienen los mismos valores de `powerpc` para los dos círculos dados.

Sintaxis

```
radical_axis(Círculo1, Círculo2)
```

Devuelve la línea cuyos puntos tienen los mismos valores de powerpc para los dos círculos dados.

0200 Radio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Radio.](#)

0779 Polinomio aleatorio

ID 779 · título recuperado de la página española

Sintaxis

```
randpoly([Var], Entero, [Intervalo, Dist])
```

Devuelve un vector de coeficientes de un polinomio de variable Var (o x), de grado Entero y donde los coeficientes son números enteros aleatorios en el rango de -99 a 99 con distribución uniforme o en un intervalo especificado por Intervalo.

Ejemplos

[Ejemplo gráfico E1] devuelve un vector de 9 números enteros aleatorios, todos ellos entre -1 y 1.

1263 range — Devuelve un vector que contiene enteros consecutivos desde Min hasta Max-1.

ID 1263 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: range

Explicación española: Devuelve un vector que contiene enteros consecutivos desde Min hasta Max-1.

Sintaxis

```
range([Min=0,] Max)
```

Devuelve un vector que contiene enteros consecutivos desde Min hasta Max-1.

1264 rand — Sin argumentos proporcionados, devuelve el siguiente número de punto flotante aleatorio en el rango de...

ID 1264 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: rand

Explicación española: Sin argumentos proporcionados, devuelve el siguiente número de punto flotante aleatorio en el rango de...

Sintaxis

```
rand([Min=1,] Máx])  
rand(Cantidad, Mín, Máx)  
rand([Cantidad=1,] V)
```

Sin argumentos proporcionados, devuelve el siguiente número de punto flotante aleatorio en el rango de 0.0 a 1.0, excluyendo 1.0.

Para un (Máx) o dos (Mín y Máx) argumentos enteros, genera un entero aleatorio del rango especificado. Por defecto, se asume que Mín es 1.

Para tres argumentos enteros, genera enteros únicos en la cantidad especificada por Cantidad, en el rango de Mín a Máx. Devuelve un vector con los números aleatorios. Dado que los números no se repiten en el resultado, Cantidad no puede ser mayor que el número de enteros disponibles en el rango de Mín a Máx. De lo contrario, lanzará un error.

Para un entero Cantidad y un vector V, seleccionará Cantidad elementos de los disponibles en el vector V sin repetición y los devolverá como un vector. Por defecto, se asume que Cantidad es 1. Si Cantidad es mayor que el número de elementos disponibles, lanzará un error.

Para esta función, puede inicializar la semilla para una aleatoriedad reproducible utilizando la función srand().

1214 Binomial aleatorio

ID 1214 · título recuperado de la página española

Sintaxis

`randbinomial(n, p)`

Devuelve un entero aleatorio con distribución binomial dados n ensayos, cada uno con una probabilidad de éxito de p.

1215 χ^2 aleatorio

ID 1215 · título recuperado de la página española

Sintaxis

`randchisquare(n)`

Devuelve un número aleatorio con distribución χ^2 dados n grados de libertad.

1216 Exponencial aleatorio

ID 1216 · título recuperado de la página española

Sintaxis

`randexp(Real)`

Dado un número real positivo, devuelve un real aleatorio según una distribución exponencial con real $a > 0$.

1217 F aleatorio

ID 1217 · título recuperado de la página española

Sintaxis

`randfisher(n, d)`

Devuelve un número aleatorio con distribución F, con numerador n y denominador d de grados de libertad.

1218 Geométrico aleatorio

ID 1218 · título recuperado de la página española

Sintaxis

`randgeometric(p)`

Devuelve un entero aleatorio con distribución geométrica dada la probabilidad p.

0573 Entero aleatorio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Entero aleatorio.](#)

1212 Permutación aleatoria

ID 1212 · título recuperado de la página española

Sintaxis

`randperm(Entero)`

Dado un entero positivo, devuelve una permutación aleatoria de $[1, 2, \dots, n]$.

Ejemplos

[Ejemplo gráfico E1] devuelve una permutación aleatoria de los elementos del vector $[1, 2, 3, 4]$

1220 Poisson aleatorio

ID 1220 · título recuperado de la página española

Sintaxis

`randpoisson(k)`

Devuelve un entero aleatorio con la distribución de Poisson dado k.

0575 Número entero aleatorio

ID 575 · título recuperado de la página española

Sintaxis

`randrange([Min=0,] Max)`

Devuelve un entero pseudoaleatorio generado en un rango de Min a Max-1.

1221 T aleatorio

ID 1221 · título recuperado de la página española

Sintaxis

`randstudent(n)`

Devuelve un número aleatorio con distribución t de Student, dado n grados de libertad.

1652 Vector aleatorio

ID 1652 · título recuperado de la página española

Sintaxis

`randvector(Enteron, Enterom)`
`randvector(Enteron, Intervalo)`
`randvector(Enteron, ley_de_distribución)`

Devuelve un vector de tamaño Enteron que contiene números enteros aleatorios en el rango de -99 a 99 (o en 0..Enterom-1) con distribución uniforme, números enteros en el Intervalo dado, o contiene números aleatorios conforme a la ley de distribución.

1213 ranm — Devuelve un vector de tamaño N o una matriz N*M que contenga números enteros aleatorios en el rango de...

ID 1213 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ranm

Explicación española: Devuelve un vector de tamaño N o una matriz N*M que contenga números enteros aleatorios en el rango de...

Sintaxis

`ranm(Entero_N [, Entero_M [, Intervalo o Distribución]])`

Devuelve un vector de tamaño N o una matriz N*M que contenga números enteros aleatorios en el rango de -99 a 99 con distribución uniforme o que contenga números aleatorios en un intervalo dado o según la distribución determinada.

Sinónimos: randmatrix, randMat

Ejemplos

[Ejemplo gráfico E1] devuelve un vector con tres elementos, cada uno de los cuales es un entero entre -99 y 99.
[Ejemplo gráfico E2]
[Ejemplo gráfico E3]

0634 Clasificación de matriz rectangular

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Clasificación de matriz rectangular.](#)

1267 ratnormal — Reescribe una expresión como una fracción racional irreducible.

ID 1267 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: ratnormal

Explicación española: Reescribe una expresión como una fracción racional irreducible.

Sintaxis

ratnormal(Expr)

Reescribe una expresión como una fracción racional irreducible.

0547 Parte real

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Parte real.](#)

0190 Reciprocación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Reciprocación.](#)

0157 Rectángulo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Rectángulo.](#)

1273 Coordenadas rectangulares

ID 1273 · título recuperado de la página española

Sintaxis

rectangular_coordinates(Vector)

Dado un vector que contiene las coordenadas polares de un punto, devuelve un vector que contiene las coordenadas rectangulares del punto.

1291 Redimensionar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Redimensionar.](#)

1248 reduced_conic — Toma una expresión cónica y devuelve un vector con los siguientes elementos

ID 1248 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: reduced_conic

Explicación española: Toma una expresión cónica y devuelve un vector con los siguientes elementos

Sintaxis

reduced_conic(Expr, [Vector])

Toma una expresión cónica y devuelve un vector con los siguientes elementos:

- El origen de la sección cónica
- La matriz de una base en la que se reduce la sección cónica
- 0 o 1 (0 si la sección cónica es degenerada)

- La ecuación reducida de la sección cónica
- Un vector de las ecuaciones paramétricas de la sección cónica

1208 Reducción de Gauss

ID 1208 · título recuperado de la página española

Sintaxis

`ref(Matriz)`

Realiza la reducción gaussiana de una matriz.

0184 Reflexión

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Reflexión.](#)

1720 Reagrupar expresión

ID 1720 · título recuperado de la página española

Sintaxis

`regroup(expr)`

Recopila los términos en una expresión. Es el equivalente a la configuración de simplificación automática de "Mínimo" en la pantalla Configuración de CAS.

0783 Resto

ID 783 · título recuperado de la página española

Sintaxis

`rem(Poli1, Poli2, [Var])`
`rem(Lista1, Lista2, [Var])`

Devuelve un vector que contiene los coeficientes del resto del cociente euclidiano de dos polinomios. Los polinomios pueden escribirse como una lista de coeficientes o en forma simbólica.

1211 Eliminar

ID 1211 · título recuperado de la página española

Sintaxis

`remove(Valor, Lista)`
`remove(Prueba, Lista)`

Dado un vector o lista, elimina las ocurrencias de Valor o elimina los valores que hacen que Prueba sea verdadero, y devuelve el vector o la lista resultante.

1210 Reordenar

ID 1210 · título recuperado de la página española

Sintaxis

`reorder(Expr, Vector)`

Dados una expresión y un vector de variables, vuelve a ordenar las variables en la expresión según el orden indicado en el vector.

1483 Reemplazar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Reemplazar.](#)

1271 **residue** — Devuelve el residuo de una expresión en el valor a.

ID 1271 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: residue

Explicación española: Devuelve el residuo de una expresión en el valor a.

Sintaxis

residue(Expr, Var, Valor)

Devuelve el residuo de una expresión en el valor a.

1258 **restart** — Depura todas las variables de CAS y restablece los ajustes de CAS.

ID 1258 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: restart

Explicación española: Depura todas las variables de CAS y restablece los ajustes de CAS.

Sintaxis

restart

Depura todas las variables de CAS y restablece los ajustes de CAS.

1230 **Resultante**

ID 1230 · título recuperado de la página española

Sintaxis

resultant(Poli1, Poli2, Var)

Devuelve la resultante (el determinante de la matriz de Sylvester) de dos polinomios.

0605 **Invertir lista**

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Invertir lista.](#)

1268 **Invertir lista**

ID 1268 · título recuperado de la página española

Sintaxis

revlist(Lista)
revlist(Vector)

Revierde el orden de los elementos de una lista o un vector.

1242 **Rombo**

ID 1242 · título recuperado de la página española

Sintaxis

rhombus(punto1, punto2, angle)

Se utiliza en la aplicación de Geometría. Dibuja un rombo, dados dos puntos y un ángulo. Como con muchos de los otros comandos del polígono, puede especificar nombres de variable opcionales CAS para almacenar las coordenadas de los otros dos vértices como puntos.

Ejemplos

[Ejemplo gráfico E1] dibuja el rombo cuyos dos vértices primeros están dados por (0, 0) y (2, 2). El ángulo en (0, 0) tiene una medida en radianes de $\pi/4$.

0156 Rombo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Rombo.](#)

0684 Lado derecho

ID 684 · título recuperado de la página española

Sintaxis

```
right(Expr1=Expr2)
right(Real1..Real2)
right(Str, n)
```

Devuelve el lado derecho de una ecuación, el extremo derecho de un intervalo o los últimos n caracteres de una cadena.

0152 Triángulo rectángulo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Triángulo rectángulo.](#)

1196 romberg — Utiliza el método de Romberg para devolver el valor aproximado de una integral definida.

ID 1196 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: romberg
Explicación española: Utiliza el método de Romberg para devolver el valor aproximado de una integral definida.

Sintaxis

```
romberg(Expr, Var, Val1, Val2)
```

Utiliza el método de Romberg para devolver el valor aproximado de una integral definida.

1283 Rotar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Rotar.](#)

0185 Rotación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Rotación.](#)

0533 Redondear

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Redondear.](#)

1259 Row

ID 1259 · título recuperado de la página española

Sintaxis

```
row(Matriz, Entero)
row(Matriz, Intervalo)
```

Dada una matriz y un entero n, devuelve la fila n de la matriz. Dada una matriz y un intervalo, devuelve un vector que contiene las filas de la matriz indicada por el intervalo.

1250 Agregar fila

ID 1250 · título recuperado de la página española

Sintaxis

```
rowAdd(Matriz, Entero1, Entero2)
```

Dados una matriz y dos números enteros, devuelve la matriz obtenida de la matriz dada después de que la fila indicada por el segundo número entero se reemplaza por la suma de las filas indicadas por los dos números enteros.

1233 Dimensión de fila

ID 1233 · título recuperado de la página española

Sintaxis

```
rowDim(Matriz)
```

Devuelve el número de filas de una matriz

1229 Norma de fila

ID 1229 · título recuperado de la página española

Sintaxis

```
rownorm(Matriz)
```

Encuentra el valor máximo (sobre todas las filas de la matriz) para la suma de los valores absolutos de todos los elementos de una fila.

1691 Subespacio de fila

ID 1691 · título recuperado de la página española

Sintaxis

```
rowspace(matriz, [variable])
```

Devuelve una matriz donde las filas son una base del espacio vectorial generado por las filas de la matriz A. Si se da, la dimensión de este espacio se almacenará en variable.

1249 Intercambiar filas

ID 1249 · título recuperado de la página española

Sintaxis

```
rowSwap(Matriz, Entero1, Entero2)
```

Dados una matriz y dos números enteros, devuelve la matriz obtenida de la matriz dada después de intercambiar las dos filas indicadas por los dos números enteros.

1254 rref — Forma escalonada reducida por filas.

ID 1254 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: rref

Explicación española: Forma escalonada reducida por filas.

Sintaxis

```
rref(Matriz)
```

```
rref(Matriz, [Entero, Opción])
```

Forma escalonada reducida por filas. Cambia una matriz rectangular a su forma escalonada reducida por filas.

1201 Solución recurrente

ID 1201 · título recuperado de la página española

Sintaxis

```
rsolve(Expr, Var, Condición)  
rsolve(Lista1, Lista2, Lista3)
```

Dada una expresión que define una relación de recurrencia, una variable y una condición inicial, devuelve la solución de forma cerrada (si es posible) de la secuencia recurrente. Dadas tres listas, cada una con varios elementos de la naturaleza anterior, resuelve el sistema de secuencias recurrentes.

0849 Funciones en vista CAS: 'S'

ID 849 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'S'

1296 SCALE — Multiplica una fila de una matriz por un escalar.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de SCALE.](#)

1292 SCALEADD — Multiplica la fila1 especificada del nombre de la matriz por valor.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de SCALEADD.](#)

0653 Descomposición Schur

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Descomposición Schur.](#)

0556 Secante

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Secante.](#)

0141 Segmento

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Segmento.](#)

1207 select

ID 1207 · título localizado en HelpEs

Sintaxis

```
select(Prueba, Lista)  
select(Prueba, Vector)
```

Dada una expresión de prueba en una única variable y una lista o vector, prueba cada elemento de la lista o del vector y devuelve una lista o un vector que contienen los elementos que satisfacen la prueba.

1251 Secuencia

ID 1251 · título recuperado de la página española

Sintaxis

```
seq(Expr, Var=Intervalo, [Incrm.])  
seq(Expr, Entero)
```

Dada una expresión, una variable definida sobre un intervalo, y un valor de incremento Opcional, devuelve un vector con la secuencia obtenida cuando se evalúa la expresión dentro del intervalo dado, usando el incremento dado. Si no se brinda incremento, el incremento usado es 1.

Dados una expresión y un entero n, devuelve una lista con la expresión repetida n veces.

1202 seqsolve

ID 1202 · título localizado en HelpEs

Sintaxis

```
seqsolve(Expr, Vector, Condición)
seqsolve(Lista1, Lista2, Lista3)
```

Similar a rsolve. Dada una expresión que define a una relación de recurrencia en términos de n o el término anterior (x), seguido por un vector de variables y una condición inicial para x (el término Décimo), retorna la solución de forma cerrada (si es posible) para la secuencia recurrente. Dadas tres listas, cada una con varios elementos de la naturaleza anterior, resuelve el sistema de secuencias recurrentes.

0689 Ampliación de serie

ID 689 · título recuperado de la página española

Sintaxis

```
series(Expr, Var=Val, [Orden], [Dir])
```

Devuelve la ampliación de serie de una expresión cercana a un valor de variable dado. Con el tercer y el cuarto argumentos opcionales puede especificar el orden y la dirección de la ampliación de la serie. Si no se especifica el orden, la serie devuelta es del quinto orden. El argumento opcional Dir es bidireccional si es 0, unilateral desde abajo si es -1, y unilateral desde arriba si es 1. Si no se especifica una dirección, la serie es bidireccional.

0548 Vector de la unidad o signo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Vector de la unidad o signo.](#)

1241 shift — Dada una lista o vector y un entero n , desplaza los elementos de la lista o vector n posiciones a la...

ID 1241 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: shift

Explicación española: Dada una lista o vector y un entero n , desplaza los elementos de la lista o vector n posiciones a la...

Sintaxis

```
shift(list [, n])
shift(vector [, n])
shift(integer [, n])
```

Dada una lista o vector y un entero n , desplaza los elementos de la lista o vector n posiciones a la izquierda si $n > 0$ o a la derecha si $n < 0$. Los elementos que salen de la lista o vector por un lado se reemplazan por 0 en el otro lado. Si no se proporciona un entero, se utiliza el valor -1, desplazando los elementos una posición a la derecha.

Dado un entero y un entero n , realiza un desplazamiento bit a bit en el primer entero; el desplazamiento es a la derecha si $n > 0$ y a la izquierda si $n < 0$. Si no se proporciona un entero, se utiliza el valor -1, desplazando los bits una posición a la izquierda.

Nótese que la dirección asociada al signo de n es opuesta para listas/vectores y enteros.

1179 Desplazamiento de fase

ID 1179 · título recuperado de la página española

Sintaxis

```
shift_phase(Expr)
```

Devuelve el resultado de aplicar un cambio de fase de $\pi/2$ a una expresión trigonométrica.

Ejemplos

[Ejemplo gráfico E1] con el ajuste de CAS Simplificar definido como Ninguna

1180 Barajar

ID 1180 · título recuperado de la página española

Sintaxis

```
shuffle([List])  
shuffle([Vector])
```

Devuelve una permutación aleatoria de una lista o vector. Esta función reorganiza aleatoriamente los elementos de la lista o vector proporcionado, barajándolos efectivamente.

Sinónimos: randperm

0671 Integral del seno

ID 671 · título recuperado de la página española

Sintaxis

```
Si(Expr)
```

Devuelve el seno integral de una expresión, $\text{int}(\sin(t)/t, t=0..x)$

1206 Firma

ID 1206 · título recuperado de la página española

Sintaxis

```
signature(Vector)
```

Devuelve la firma de una permutación dada como un vector.

0187 Similitud

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Similitud.](#)

0674 Simplificar expresión

ID 674 · título recuperado de la página española

Sintaxis

```
simplify(Expr)
```

Simplifica una expresión.

1685 Reducción simple

ID 1685 · título recuperado de la página española

Sintaxis

```
simplex_reduce(Matrix_A, Vector_B, Vector_C)  
simplex_reduce(Matrix)
```

Aplica el algoritmo símplex para maximizar $c \cdot x$ sujeto a $A \cdot x \leq b$, $x \geq 0$, $b \geq 0$.

Matrix_A es la matriz de restricciones A.

Vector_B es el vector del lado derecho b.

Vector_C es el vector de coeficientes objetivo c.

Devuelve el valor máximo, el vector solución aumentado x y la matriz reducida.

También se puede llamar a simplex_reduce con un único argumento matricial de la forma $[[A||b],[-c|*|0]]$.

1247 simult — Devuelve la solución a un sistema de ecuaciones lineales o varios sistemas de ecuaciones lineales...

ID 1247 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `simult`

Explicación española: Devuelve la solución a un sistema de ecuaciones lineales o varios sistemas de ecuaciones lineales...

Sintaxis

`simult(Matriz1, Matriz2)`

Devuelve la solución a un sistema de ecuaciones lineales o varios sistemas de ecuaciones lineales presentadas en forma de matriz. En el caso de un sistema de ecuaciones lineales, toma una matriz de coeficientes y una matriz de columna de constantes, y devuelve la matriz de columna de la solución.

1274 Seno

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Seno.](#)

0731 sin2costan — Reescribe Expr de forma que $\sin(x)$ se reemplaza por $\cos(x)\tan(x)$

ID 731 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `sin2costan`

Explicación española: Reescribe Expr de forma que $\sin(x)$ se reemplaza por $\cos(x)\tan(x)$

Sintaxis

`sin2costan(Expr)`

Reescribe Expr de forma que $\sin(x)$ se reemplaza por $\cos(x)\tan(x)$

0561 Seno hiperbólico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Seno hiperbólico.](#)

1187 sincos — Devuelve una expresión con los exponenciales complejos reescritos en términos de seno y coseno.

ID 1187 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `sincos`

Explicación española: Devuelve una expresión con los exponenciales complejos reescritos en términos de seno y coseno.

Sintaxis

`sincos(Expr)`

Devuelve una expresión con los exponenciales complejos reescritos en términos de seno y coseno.

0135 Intersección única

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Intersección única.](#)

0608 Tamaño de la lista

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tamaño de la lista.](#)

0204 Slope

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Slope.](#)

1175 slopeat — Se utiliza en la Vista simbólica de la aplicación de Geometría.

ID 1175 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: slopeat

Explicación española: Se utiliza en la Vista simbólica de la aplicación de Geometría.

Sintaxis

```
slopeat(Segmento, Punto)
slopeat(Línea, Punto)
slopeat(Raya, Punto)
```

Se utiliza en la Vista simbólica de la aplicación de Geometría. Muestra la pendiente de un objeto recto (segmento, línea, etc.). La medida se muestra, con una etiqueta, en el momento dado en la Vista de gráfico.

Ejemplos

En la Vista simbólica, `slopeat(line(point(0,0), point(2,3)), point(-8,8))` muestra “`sline(point(0,0), point(2,3))=3/2`” en el punto $(-8, 8)$ en la Vista de gráfico.

1195 Orden.

ID 1195 · título recuperado de la página española

Sintaxis

```
sort(List)
sort(Vector)
sort(Obj_1, ..., Obj_N)
```

Ordena una lista, un vector o una secuencia de objetos. Si los objetos ordenados son números o cadenas, el orden se realiza en orden ascendente. Si los objetos ordenados son expresiones, entonces el orden se realiza en orden ascendente de valores numéricos, sumas y productos, en orden exponencial ascendente.

0632 Radio espectral

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Radio espectral.](#)

0631 Norma espectral

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Norma espectral.](#)

0710 Solución de CAS

ID 710 · título recuperado de la página española

Sintaxis

```
solve(Expr [, Var])
solve({Eq1, Eq2, ...} [, Var])
solve(Expr, Var=Guess)
solve(Expr, Var=Val1..Val2)
```

Devuelve una lista de las soluciones (reales y complejas) a una ecuación polinómica, un conjunto de ecuaciones polinómicas o desigualdades. Por defecto, resuelve para $\text{Var}=x$ si x está en Expr , de lo contrario resuelve para la primera variable encontrada de izquierda a derecha. La configuración de CAS afectará las soluciones proporcionadas.

Se aconseja al usuario que proporcione una conjetura o defina un intervalo en el que buscar una solución para obtener los mejores resultados en los casos en que se sabe que la solución es aproximada. Para proporcionar una conjetura, use la sintaxis $\text{Var}=\text{Guess}$. Para proporcionar un intervalo, use la sintaxis $\text{Var}=\text{Val1}..\text{Val2}$. En este último caso, la búsqueda se limita al intervalo cerrado $[\text{Val1}, \text{Val2}]$.

0604 Ordenar lista

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ordenar lista.](#)

1269 Spline

ID 1269 · título recuperado de la página española

Sintaxis

```
spline(ListaX, ListaY, Var, Entero)
spline(VectorX, VectorY, Var, Entero)
```

Dadas dos listas o vectores (uno para los valores x y uno de los valores y), así como una variable y un grado entero, devuelve el spline natural a través de los puntos dados por dos listas. Los polinomios en el spline están de acuerdo a la variable dada y son de grado dado.

1270 Dividir

ID 1270 · título recuperado de la página española

Sintaxis

```
split(Str, Sep)
split(Expr, Vars)
```

Para una cadena Str , divide la cadena en una secuencia de subcadenas utilizando el separador especificado Sep . Para una expresión $Expr$, intenta factorizar la expresión que involucra dos variables (especificadas en el vector $Vars$) en el producto de dos expresiones, cada una dependiente solo de una de esas variables.

1747 Sq — Devuelve el cuadrado del valor `n`.

ID 1747 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `sq`
Explicación española: Devuelve el cuadrado del valor `n``.

Sintaxis

```
sq(n)
```

Devuelve el cuadrado del valor n .

1234 sqrfree — Devuelve un polinomio factorizado como un producto de potencias de factores coprimos donde cada factor...

ID 1234 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `sqrfree`
Explicación española: Devuelve un polinomio factorizado como un producto de potencias de factores coprimos donde cada factor...

Sintaxis

```
sqrfree(Expr)
```

Devuelve un polinomio factorizado como un producto de potencias de factores coprimos donde cada factor tiene raíces de multiplicidad 1

1189 Raíz cuadrada

ID 1189 · título recuperado de la página española

Sintaxis

`sqrt(Expr)`
`√(Expr)`

Devuelve la raíz cuadrada de Expr. Acepta valores numéricos, listas, vectores y matrices. En la vista CAS, también acepta expresiones simbólicas.

0154 Cuadrado

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Cuadrado.](#)

1265 srand — La función se utiliza para inicializar el generador de números aleatorios empleado por las funciones...

ID 1265 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `srand`

Explicación española: La función se utiliza para inicializar el generador de números aleatorios empleado por las funciones...

Sintaxis

`srand(seed)`

La función se utiliza para inicializar el generador de números aleatorios empleado por las funciones `rand()` / `randint()` / `randrange()` / `randvector()` [...].

El argumento `seed` establece el punto de partida del generador, lo que permite que los resultados producidos por las funciones aleatorias sean reproducibles si se vuelven a llamar con la misma `seed`. Esto significa que usar la misma `seed` producirá las mismas secuencias de números aleatorios en llamadas posteriores.

Al aleatorizar vectores completos, los elementos dentro del vector permanecerán iguales si se usa la misma `seed`, pero pueden devolverse en un orden diferente.

1235 Desviación estándar de población

ID 1235 · título recuperado de la página española

Sintaxis

`stddev(Lista1, [Lista2])`
`stddev(Vector1, [Vector2])`
`stddev(Matriz)`

Devuelve la desviación estándar de la población de los elementos de una lista o vector, o una lista de las desviaciones estándar de la población de las columnas de una matriz. La segunda lista opcional es una lista de pesos.

1236 Desviación estándar de muestra

ID 1236 · título recuperado de la página española

Sintaxis

`stddevp(Lista1, [Lista2])`
`stddevp(Vector1, [Vector2])`
`stddevp(Matriz)`

Devuelve la desviación estándar de los elementos de una lista o vector, o de una lista de desviaciones estándar de las columnas de una matriz. La segunda lista opcional es una lista de pesos.

1188 Almacenar

ID 1188 · título recuperado de la página española

Sintaxis

`sto(Obj, Var)`

Almacena el objeto dado como primer argumento en la variable dada como segundo argumento.

1192 strip — Devuelve una copia de la cadena `Str` con los caracteres iniciales y finales especificados eliminados.

ID 1192 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: strip

Explicación española: Devuelve una copia de la cadena `Str` con los caracteres iniciales y finales especificados eliminados.

Sintaxis

strip(Str, Chars)

Devuelve una copia de la cadena Str con los caracteres iniciales y finales especificados eliminados. El argumento Chars es una cadena que especifica el conjunto de caracteres a eliminar. Si se omite, el argumento Chars elimina por defecto los espacios en blanco ASCII. El argumento Chars no es un prefijo o sufijo; más bien, todas las combinaciones de sus valores se eliminan tanto del principio como del final de la cadena.

0579 Densidad T de Student

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Densidad T de Student.](#)

1193 sturm — La función sturm toma dos o cuatro argumentos y calcula las secuencias de Sturm y las propiedades de las...

ID 1193 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: sturm

Explicación española: La función sturm toma dos o cuatro argumentos y calcula las secuencias de Sturm y las propiedades de las...

Sintaxis

sturm(Pólino, Variable)
sturm(Pólino, Variable, a, b)

La función sturm toma dos o cuatro argumentos y calcula las secuencias de Sturm y las propiedades de las raíces de un polinomio o expresión racional:

- Dos argumentos: Cuando se proporciona un polinomio Pólino o una fracción racional P/Q, y un nombre de variable Variable, sturm devuelve la lista de secuencias de Sturm junto con las multiplicidades de los factores sin cuadrado de Pólino (o P/Q). Este modo es similar a la función sturmseq.
- Cuatro argumentos: Cuando se proporciona un polinomio Pólino, un nombre de variable Variable, y dos números a y b (ya sean reales o complejos): Si a y b son reales, sturm calcula y devuelve el número de cambios de signo en la secuencia de Sturm de Pólino evaluada donde $a < \text{Variable} \leq b$. Si a o b es complejo, sturm calcula y devuelve el número de raíces complejas de Pólino dentro del rectángulo definido por a y b como vértices opuestos.

0787 Recuento de ceros

ID 787 · título recuperado de la página española

Sintaxis

sturmab(Poli [, Var], Intervalo)

Si el intervalo es real, devuelve el número de cambios de signo en el polinomio especificado en el intervalo. Si el intervalo es complejo, devuelve el número de raíces complejas en el rectángulo limitado por el intervalo. Si se omite Var, se asume que es x.

1194 sturmseq — Devuelve la secuencia de Sturm correspondiente a un polinomio o a una fracción racional.

ID 1194 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: sturmseq

Explicación española: Devuelve la secuencia de Sturm correspondiente a un polinomio o a una fracción racional.

Sintaxis

```
sturmseq(Poli, [Var])  
sturmseq(RatFrac, [Var])
```

Devuelve la secuencia de Sturm correspondiente a un polinomio o a una fracción racional.

1293 Extraer parte

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Extraer parte.](#)

1256 Sub matriz

ID 1256 · título recuperado de la página española

Sintaxis

```
subMat(Matriz, Ent1, Ent2, Ent3, Ent4)
```

Extractos de una sub matriz de la matriz cuya diagonal se define por cuatro enteros. Los dos primeros enteros definen la fila y la columna del primer elemento y los dos últimos enteros definen la fila y columna del último elemento de la sub matriz.

0679 Sustituir

ID 679 · título recuperado de la página española

Sintaxis

```
subst(Expr, Var=valor)
```

Sustituye un valor para una variable en una expresión

0690 Sumatorio

ID 690 · título recuperado de la página española

Sintaxis

```
sum(List)  
sum(Vector)  
sum(Expr, Var [, Real1, Real2 [, Step]])
```

En su forma básica, cuando se proporciona solo un argumento, devuelve la suma de todos los elementos en la lista dada List o en el vector Vector.

Con solo los dos primeros argumentos, devuelve la antiderivada discreta de la expresión Expr con respecto a la variable Var.

En una forma más avanzada, devuelve la suma discreta de Expr con respecto a la variable Var desde Real1 hasta Real2, utilizando el paso opcional Step.

0703 Suma de Riemann

ID 703 · título recuperado de la página española

Sintaxis

```
sum_riemann(Expr(Xpr), Lst(var1, var2))
```

Devuelve, en las proximidades de $n=+\infty$ un equivalente de la suma de $Xpr(var1, var2)$ para $var2$ desde $var2=1$ a $var2=var1$ si la suma se considera una suma de Riemann asociada con la función continua definida en $[0,1]$.

1204 Suprimir

ID 1204 · título recuperado de la página española

Sintaxis

```
suppress(object, index)
suppress(object, start, end)
suppress(object, {index1, index2, ..., indexN})
suppress(string1, string2)
```

Elimina elementos de `object` mediante un único índice de posición, un intervalo especificado por los índices inicial y final, o una lista de índices. `object` puede ser una lista, un vector o una cadena.

En el caso de `string1` y `string2`, se elimina de `string1` cada aparición de los caracteres contenidos en `string2`.

En las operaciones con cadenas, los caracteres no ASCII no se procesan de forma fiable y pueden producir resultados incorrectos.

1205 Surd

ID 1205 · título recuperado de la página española

Sintaxis

```
surd(Expr, Entero)
```

Dados una expresión y un número entero `n`, devuelve la expresión elevada a la potencia $1/n$.

0655 Valores únicos

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Valores únicos.](#)

1294 Intercambiar columnas

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Intercambiar columnas.](#)

1295 Intercambiar filas

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Intercambiar filas.](#)

1209 Matriz de Sylvester

ID 1209 · título recuperado de la página española

Sintaxis

```
sylvester(Poli, Poli, Var)
```

Devuelve la matriz de Sylvester de dos polinomios

0775 Polinomio a coeficientes

ID 775 · título recuperado de la página española

Sintaxis

```
symb2poly(Expr [, Var])
symb2poly(Expr, {Var1, ..., VarN})
```

Dado un polinomio, devuelve un vector que contiene los coeficientes del polinomio. Con una variable como segundo argumento, devuelve los coeficientes de un polinomio con respecto a dicha variable. Con una lista de variables como segundo argumento, devuelve el formato interno del polinomio. Es, esencialmente, la inversa de `poly2symb()`.

0850 Funciones en vista CAS: 'T'

ID 850 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'T'

1255 Tabla CAS

ID 1255 · título recuperado de la página española

Sintaxis

```
table(SeqEqual(Índice=valor))
```

Define una matriz donde el índice son cadenas de caracteres o números reales

1713 Tabla de variación

ID 1713 · título recuperado de la página española

Sintaxis

```
tabvar(function [, variable [, minimum, maximum]])  
tabvar(function [, variable=min..max])
```

Crea una tabla de variación para la función dada. Si no se especifican los límites del intervalo (minimum y maximum), se asumen valores predeterminados. Si no se indica la variable de la función (variable), también se asume una variable predeterminada.

1191 tail — Dado un vector, una lista, una cadena o una secuencia de objetos, devuelve un vector con el primer...

ID 1191 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: tail

Explicación española: Dado un vector, una lista, una cadena o una secuencia de objetos, devuelve un vector con el primer...

Sintaxis

```
tail(Vector)  
tail(Lista)  
tail(Cadena)  
tail(Obj1, Obj2, ...)
```

Dado un vector, una lista, una cadena o una secuencia de objetos, devuelve un vector con el primer elemento eliminado.

1414 Tangente

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tangente](#).

0565 Tangente hiperbólica

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tangente hiperbólica](#).

1185 tan2cossin2 — Devuelve una expresión con tan(x) reescrita como (1-cos(2*x))/sin(2*x).

ID 1185 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: tan2cossin2

Explicación española: Devuelve una expresión con tan(x) reescrita como (1-cos(2*x))/sin(2*x).

Sintaxis

```
tan2cossin2(Expr)
```

Devuelve una expresión con tan(x) reescrita como (1-cos(2*x))/sin(2*x).

0739 tan2sincos — Reescribe Expr con tan(x) usando sin(x)/cos(x)

ID 739 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: tan2sincos

Explicación española: Reescribe Expr con $\tan(x)$ usando $\sin(x)/\cos(x)$

Sintaxis

`tan2sincos(Expr)`

Reescribe Expr con $\tan(x)$ usando $\sin(x)/\cos(x)$

1186 tan2sincos2 — Devuelve una expresión con $\tan(x)$ reescrita como $\sin(2*x)/(1+\cos(2*x))$.

ID 1186 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: tan2sincos2

Explicación española: Devuelve una expresión con $\tan(x)$ reescrita como $\sin(2*x)/(1+\cos(2*x))$.

Sintaxis

`tan2sincos2(Expr)`

Devuelve una expresión con $\tan(x)$ reescrita como $\sin(2*x)/(1+\cos(2*x))$.

0146 Tangente

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tangente.](#)

0701 Ampliación de serie de Taylor

ID 701 · título recuperado de la página española

Sintaxis

`taylor(Expr [, Var=Value [, Order]] [, polynom])`

Devuelve la expansión en serie de Taylor de una expresión en un punto o en el infinito (por defecto, en $x=0$ y con un orden relativo de 5). El término de resto `order_size()` se puede eliminar utilizando el argumento opcional `polynom`.

Ejemplos

[Ejemplo gráfico E1]
[Ejemplo gráfico E2]
[Ejemplo gráfico E3]

0797 Tn de Chebyshev

ID 797 · título recuperado de la página española

Sintaxis

`tchebyshev1(Entero)`

Devuelve el polinomio de Tchebyshev del primer tipo de grado n.

0798 Un de Chebyshev

ID 798 · título recuperado de la página española

Sintaxis

`tchebyshev2(Entero)`

Devuelve el polinomio de Tchebyshev del segundo tipo de grado n.

1714 tcoeff — Devuelve el coeficiente del término de menor grado de un polinomio.

ID 1714 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `tcoeff`

Explicación española: Devuelve el coeficiente del término de menor grado de un polinomio.

Sintaxis

```
tcoeff(Poli)
tcoeff(Vector)
tcoeff(Lista)
```

Devuelve el coeficiente del término de menor grado de un polinomio. El polinomio se puede expresar en forma simbólica o como un vector o lista de coeficientes.

0748 tcollect — Devuelve una expresión trigonométrica linealizada y con cualquier término de seno y coseno del mismo...

ID 748 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `tcollect`

Explicación española: Devuelve una expresión trigonométrica linealizada y con cualquier término de seno y coseno del mismo...

Sintaxis

```
tcollect(Expr)
```

Devuelve una expresión trigonométrica linealizada y con cualquier término de seno y coseno del mismo ángulo agrupados..

0722 Ampliación trascendental

ID 722 · título recuperado de la página española

Sintaxis

```
texpand(Expr)
```

Amplía una expresión trascendental; es decir, una expresión que contiene funciones trigonométricas, logarítmicas o exponenciales.

`texpand` desarrolla la expresión en términos de `sin()`, `cos()`, `ln()`, y `exp()`.

1748 time — Cuando se proporciona el argumento ``Cmd``, mide el tiempo que tarda en ejecutarse el comando especificado.

ID 1748 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `time`

Explicación española: Cuando se proporciona el argumento ``Cmd``, mide el tiempo que tarda en ejecutarse el comando especificado.

Sintaxis

```
time(Cmd)
time()
```

Cuando se proporciona el argumento `Cmd`, mide el tiempo que tarda en ejecutarse el comando especificado.

Sin argumentos, devuelve una marca de tiempo que representa el valor de un reloj monótono en segundos, con una precisión de 1 milisegundo.

0747 tlin — Devuelve una expresión trigonométrica con los productos y las potencias de enteros linealizados.

ID 747 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `tlin`

Explicación española: Devuelve una expresión trigonométrica con los productos y las potencias de enteros linealizados.

Sintaxis

`tlin(Expr)`

Devuelve una expresión trigonométrica con los productos y las potencias de enteros linealizados.

0138 Trazar**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar.](#)

0636 Trazar**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar.](#)

0183 Traslación**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Traslación.](#)

1200 Matriz de transposición

ID 1200 · título recuperado de la página española

Sintaxis

`transpose(Matriz)`

Transpone una matriz (sin conjugación).

0150 Triángulo**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Triángulo.](#)

0750 trig2exp — Devuelve una expresión con las funciones trigonométricas reescritas como complejos exponenciales (sin...

ID 750 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `trig2exp`

Explicación española: Devuelve una expresión con las funciones trigonométricas reescritas como complejos exponenciales (sin...

Sintaxis

`trig2exp(Expr)`

Devuelve una expresión con las funciones trigonométricas reescritas como complejos exponenciales (sin linealización).

0744 trigcos — Devuelve una expresión simplificada mediante las fórmulas $\sin(x)^2 + \cos(x)^2 = 1$ y $\tan(x) = \sin(x)/\cos(x)$.

ID 744 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `trigcos`

Explicación española: Devuelve una expresión simplificada mediante las fórmulas $\sin(x)^2 + \cos(x)^2 = 1$ y $\tan(x) = \sin(x)/\cos(x)$.

Sintaxis

trigcos(Expr)

Devuelve una expresión simplificada mediante las fórmulas $\sin(x)^2 + \cos(x)^2 = 1$ y $\tan(x) = \sin(x)/\cos(x)$. Se da prioridad a $\cos(x)$ por sobre $\sin(x)$ y $\tan(x)$ en el resultado.

0749 trigexpand — Devuelve una expresión trigonométrica en forma expandida.

ID 749 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: trigexpand

Explicación española: Devuelve una expresión trigonométrica en forma expandida.

Sintaxis

trigexpand(Expr)

Devuelve una expresión trigonométrica en forma expandida.

0743 trigsin — Devuelve una expresión simplificada mediante las fórmulas $\sin(x)^2 + \cos(x)^2 = 1$ y $\tan(x) = \sin(x)/\cos(x)$.

ID 743 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: trigsin

Explicación española: Devuelve una expresión simplificada mediante las fórmulas $\sin(x)^2 + \cos(x)^2 = 1$ y $\tan(x) = \sin(x)/\cos(x)$.

Sintaxis

trigsin(Expr)

Devuelve una expresión simplificada mediante las fórmulas $\sin(x)^2 + \cos(x)^2 = 1$ y $\tan(x) = \sin(x)/\cos(x)$. Se da prioridad a $\sin(x)$ por sobre $\cos(x)$ y $\tan(x)$ en el resultado.

0745 trigtan — Devuelve una expresión simplificada mediante las fórmulas $\sin(x)^2 + \cos(x)^2 = 1$ y $\tan(x) = \sin(x)/\cos(x)$ Se da...

ID 745 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: trigtan

Explicación española: Devuelve una expresión simplificada mediante las fórmulas $\sin(x)^2 + \cos(x)^2 = 1$ y $\tan(x) = \sin(x)/\cos(x)$ Se da...

Sintaxis

trigtan(Expr)

Devuelve una expresión simplificada mediante las fórmulas $\sin(x)^2 + \cos(x)^2 = 1$ y $\tan(x) = \sin(x)/\cos(x)$ Se da prioridad a $\tan(x)$ por sobre $\sin(x)$ y $\cos(x)$ en el resultado.

0616 Transposición**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Transposición.](#)

1182 trunc — Dado un valor o una lista de valores, así como un número entero n, devuelve el valor o la lista truncado...

ID 1182 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: trunc

Explicación española: Dado un valor o una lista de valores, así como un número entero n, devuelve el valor o la lista truncado...

Sintaxis

```
trunc(Real, [Entero])  
trunc(Lista, [Entero])
```

Dado un valor o una lista de valores, así como un número entero n , devuelve el valor o la lista truncado en n decimales. Si no se proporciona n , se toma como 0. Este comando también acepta números complejos.

1199 Truncar

ID 1199 · título recuperado de la página española

Sintaxis

```
truncate(Poli, Entero)
```

Dado un polinomio y un entero n , trunca el polinomio de orden n .

1184 Simplificar trascendental

ID 1184 · título recuperado de la página española

Sintaxis

```
tsimplify(Expr)
```

Devuelve una expresión con reescritura trascendental como exponenciales complejos

1198 type — Devuelve el código de tipo del objeto CAS especificado.

ID 1198 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: type

Explicación española: Devuelve el código de tipo del objeto CAS especificado.

Sintaxis

```
type(Object)
```

Devuelve el código de tipo del objeto CAS especificado.

- 0: DOM_int
- 1: DOM_FLOAT
- 2: DOM_INT
- 3: DOM_LONGFLOAT
- 4: DOM_COMPLEX
- 5: (No disponible - polinomio multivariado)
- 6: DOM_IDENT
- 7: DOM_LIST
- 8: DOM_SYMBOLIC
- 9: (No disponible - expansión de serie univariada dispersa)
- 10: DOM_RAT
- 11: (No disponible - extensión algebraica)
- 12: DOM_STRING
- 13: DOM_FUNC
- 14: (No disponible - real_complex_rootof)
- 15: type(a %% n) - mod
- 16: type(GF(q)) - user
- 17: DOM_MAP
- (21): DOM_SPECIALFLOAT

0851 Funciones en vista CAS: 'U'

ID 851 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'U'

1310 Conversión de factores de unidades

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Conversión de factores de unidades.](#)

1320 unapply — Devuelve la función definida por una expresión y una variable.

ID 1320 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: unapply

Explicación española: Devuelve la función definida por una expresión y una variable.

Sintaxis

unapply(Expr, Var)

Devuelve la función definida por una expresión y una variable.

1307 Uniforme continua

ID 1307 · título recuperado de la página española

Sintaxis

uniform(a, b, x)

Función de densidad de probabilidad uniforme

Calcula la densidad de probabilidad de la distribución uniforme a x, con los parámetros a y b.

1308 Uniforme acumulada

ID 1308 · título recuperado de la página española

Sintaxis

uniform_cdf(a, b, x, [x2])

Función de distribución uniforme acumulada

Devuelve la probabilidad de cola inferior de la función de densidad de probabilidad uniforme para el valor x , con los parámetros a y b.

1309 Uniforme acumulada inversa

ID 1309 · título recuperado de la página española

Sintaxis

uniform_icdf(a, b, p)

Función de distribución uniforme acumulada inversa

Devuelve el valor de x de forma tal que la probabilidad de cola inferior uniforme de x, con los parámetros a y b, sea p.

1312 Mayúsculas

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Mayúsculas.](#)

1311 Simplificación de unidades

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Simplificación de unidades.](#)

0852 Funciones en vista CAS: 'V'

ID 852 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'V'

1316 Valuación

ID 1316 · título recuperado de la página española

Sintaxis

`valuation(Poli, [Var])`

Devuelve el valor (grado del término de menor grado) de un polinomio. Con solo un polinomio como argumento, devuelve el valor para x . Con una variable como segundo argumento, se realiza el valor para esta.

0626 Matriz de Vandermonde

ID 626 · título recuperado de la página española

Sintaxis

`vandermonde(vector)`

Dado un vector $[n_1, n_2, \dots, n_j]$, devuelve una matriz cuya primera fila es $[(n_1)^0, (n_1)^1, (n_1)^2, \dots, (n_1)^{(j-1)}]$. La segunda fila es $[(n_2)^0, (n_2)^1, (n_2)^2, \dots, (n_2)^{(j-1)}]$, etc.

1319 Varianza

ID 1319 · título recuperado de la página española

Sintaxis

`variance(Lista1, [Lista2])`
`variance(Matriz)`

Devuelve la varianza de una lista o la lista de varianzas de las columnas de una matriz. La segunda lista opcional es una lista de pesos.

1324 vector — Dado un punto, define un vector desde el origen hasta el punto dado.

ID 1324 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `vector`

Explicación española: Dado un punto, define un vector desde el origen hasta el punto dado.

Sintaxis

`vector(Punto1, [Punto2])`

Dado un punto, define un vector desde el origen hasta el punto dado. Con dos puntos, define el vector desde el primer punto al segundo.

Ejemplos

[Ejemplo gráfico E1] crea un vector de (1,1) a (3,0).

1318 Vértices

ID 1318 · título recuperado de la página española

Sintaxis

`vertices(Polígono)`

Devuelve un vector que contiene la lista de los vértices de un polígono.

1317 Vértices cerrados

ID 1317 · título recuperado de la página española

Sintaxis

vertices_abca(Polígono)

Devuelve un vector que contiene la lista cerrada de los vértices de un polígono.

1315 vpotential — Dados un vector **V** y un vector de variables, devuelve el vector **U** tal que $\text{curl}(\mathbf{U})=\mathbf{V}$.

ID 1315 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: vpotential

Explicación española: Dados un vector **V** y un vector de variables, devuelve el vector **U** tal que $\text{curl}(\mathbf{U})=\mathbf{V}$.

Sintaxis

vpotential(Vector1, Vector2)

Dados un vector **V** y un vector de variables, devuelve el vector **U** tal que $\text{curl}(\mathbf{U})=\mathbf{V}$.

0853 Funciones en vista CAS: 'W'

ID 853 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'W'

1325 Weibull continuo

ID 1325 · título recuperado de la página española

Sintaxis

weibull(k, n, [t], x)

Función de densidad de probabilidad de Weibull

Calcula la densidad de probabilidad de la distribución de Weibull en **x**, con los parámetros **k**, **n** y **t**. De forma predeterminada, **t** es 0.

1326 Weibull acumulada

ID 1326 · título recuperado de la página española

Sintaxis

weibull_cdf(k, n, [t], x, [x2])

Función de distribución de Weibull acumulada

Devuelve la probabilidad de cola inferior de la función de densidad de probabilidad de Weibull para el valor **x**, con los parámetros **k**, **n** y **t**. De forma predeterminada, **t** es 0.

1327 Weibull acumulada inversa

ID 1327 · título recuperado de la página española

Sintaxis

weibull_icdf(k, n, [t], x)

Función de distribución de Weibull acumulada inversa

Devuelve el valor **x** de forma que la probabilidad de cola más baja de Weibull de **x**, dados los parámetros **k**, **n** y **t**

1314 Condición When (cuando)

ID 1314 · título recuperado de la página española

Sintaxis

`when(Cond, Expr1, Expr2)`

Si la condición es verdadera, devuelve Expr1; de lo contrario, devuelve Expr2.

1655 Distribución de Wilcoxon

ID 1655 · título recuperado de la página española

Sintaxis

`wilcoxonp(Entero1, [Entero2])`

Prueba de distribución de Wilcoxon o Mann-Whitney para una o dos muestras.

1656 wilcoxons — Establece estadísticas de Wilcoxon o Mann-Whitney para una muestra (Lista1) y Mediana, o dos muestras...

ID 1656 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: `wilcoxons`

Explicación española: Establece estadísticas de Wilcoxon o Mann-Whitney para una muestra (Lista1) y Mediana, o dos muestras...

Sintaxis

`wilcoxons(Lista1, Mediana)`
`wilcoxons(Lista1, Lista2)`

Establece estadísticas de Wilcoxon o Mann-Whitney para una muestra (Lista1) y Mediana, o dos muestras (Lista1,Lista2).

1657 Prueba de wilcoxon

ID 1657 · título recuperado de la página española

Sintaxis

`wilcoxont(Lista1, Mediana, [Método], [Significado])`
`wilcoxont(Lista1, Lista2)`
`wilcoxont(Lista, Lista || Real, [Func], [Real])`

Prueba de Wilcoxon o Mann-Whitney para una muestra (Lista1) y Mediana, o dos muestras (Lista1,Lista2). De manera opcional, se especifica Método como '<' o '>', y Significado.

0854 Funciones en vista CAS: 'X'

ID 854 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'X'

0536 Exponente

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Exponente.](#)

0855 Funciones en vista CAS: 'Z'

ID 855 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con 'Z'

0668 Zeta — Devuelve el valor de la función zeta (Z) para un número x real.

ID 668 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: Zeta

Explicación española: Devuelve el valor de la función zeta (Z) para un número x real.

Sintaxis

Zeta(x)

Devuelve el valor de la función zeta (Z) para un número x real.

1323 zip — Aplica una función de dos variables a los elementos de dos listas o vectores y devuelve el resultado en...

ID 1323 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: zip

Explicación española: Aplica una función de dos variables a los elementos de dos listas o vectores y devuelve el resultado en...

Sintaxis

zip('Función', Lista1, Lista2, Predeterminado)

zip('Función', Vector1, Vector2, Predeterminado)

Aplica una función de dos variables a los elementos de dos listas o vectores y devuelve el resultado en un vector. Sin el valor predeterminado, la longitud del vector es el mínimo de las longitudes de las dos listas. Con el valor predeterminado, la lista más corta se completa con el valor predeterminado.

1313 Transformar Z

ID 1313 · título recuperado de la página española

Sintaxis

ztrans(Expr, [Var], [ZtransVar])

Devuelve la transformación Z de una secuencia.

0711 Ceros

ID 711 · título recuperado de la página española

Sintaxis

zeros(Expr [, Var])

zeros(Exprs, Vars)

El comando zeros devuelve los ceros de la expresión especificada por Expr para la variable especificada por Var. Los ceros devueltos pueden ser reales o complejos, según la configuración del CAS. De esta forma, el resultado es la solución de la ecuación $\text{Expr} = 0$.

El comando zeros también puede resolver un sistema de ecuaciones. De esta forma, Exprs especifica una lista o vector de expresiones o ecuaciones, y Vars especifica la lista o vector correspondiente de variables a resolver. El resultado es una matriz cuyas filas son las soluciones del sistema.

Ejemplos

[Ejemplo gráfico E1]

[Ejemplo gráfico E2] si la opción Use i está marcada en la configuración del CAS y [] en caso contrario.

0856 Funciones en vista CAS: símbolos

ID 856 · título recuperado de la página española

Lista de funciones y comandos en la vista CAS que comienzan con símbolos

1436 Tecla igual

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tecla igual.](#)

0542 Porcentaje

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Porcentaje.](#)

1343 Porcentaje de cambio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Porcentaje de cambio.](#)

1344 Porcentaje Total

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Porcentaje Total.](#)

0857 Índice de comandos Python

ID 857 · título localizado en HelpEs

Lista de todas las funciones y comandos disponibles en Python

0858 Funciones en Python: 'A'

ID 858 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'A'

1895 abs (builtins) — Devuelve el valor absoluto de un número.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de abs \(builtins\)](#).

2113 abs (linalg) — Devuelve una matriz que contiene el valor absoluto de x, elemento por elemento.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de abs \(linalg\)](#).

2148 acos (math) — Devuelve el coseno inverso de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de acos \(math\)](#).

2142 acosh (math) — Devuelve el coseno hiperbólico inverso de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de acosh \(math\)](#).

1839 add (builtins.set) — Agrega el elemento especificado `e` al conjunto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de add \(builtins.set\)](#).

2108 add (linalg) — Devuelve un vector o matriz que contiene la suma de A y B, elemento por elemento.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de add \(linalg\)](#).

1896 all (builtins) — Devuelve True si todos los elementos del iterable `i` son verdaderos, o si `i` está vacío.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de all \(builtins\)](#).

1897 any (builtins) — Devuelve True si algún elemento del iterable `i` es verdadero.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de any \(builtins\)](#).

1787 append (builtins bytearray) — Agrega un solo byte b (un entero entre 0 y 255) al final del bytearray.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de append \(builtins bytearray\)](#).

1818 append (builtins.list) — Agrega el elemento x al final de la lista.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de append \(builtins.list\)](#).

1980 append (array.array) — Agrega el elemento especificado x al final del arreglo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de append \(array.array\).](#)

2026 append (ucollections) — Agrega el elemento `x` al extremo derecho del objeto deque.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de append \(ucollections\).](#)

2107 apply (linalg) — Aplica la función `f` elemento a elemento a `m`, donde `m` es una tupla, una lista o una matriz.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de apply \(linalg\).](#)

2084 arange (linalg) — Con un solo argumento entero, devuelve una secuencia de valores de [0, a-1].

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de arange \(linalg\).](#)

2256 arc (hprime) — Dibuja un arco de un círculo en el `gráfico` especificado, centrado en (`x`, `y`), con un `radio` y un...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de arc \(hprime\).](#)

2272 arc_c (hprime) — Dibuja un arco de un círculo en el `gráfico` especificado, centrado en las coordenadas cartesianas (`x`, ...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de arc_c \(hprime\).](#)

2185 argv (sys) — Una lista mutable de argumentos con los que se inició el programa actual.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de argv \(sys\).](#)

1979 array, clase (array)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de array, clase \(array\).](#)

1980 append (array.array) — Agrega el elemento especificado x al final del arreglo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de append \(array.array\).](#)

1981 extend (array.array) — Extiende el arreglo agregando todos los elementos del iterable proporcionado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de extend \(array.array\).](#)

2078 asc (arit) — Devuelve una versión legible de cualquier objeto (cadenas, tuplas, listas, etc.).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de asc \(arit\).](#)

2149 asin (math) — Devuelve el arco seno de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de asin \(math\).](#)

2143 asinh (math) — Devuelve el seno hiperbólico inverso de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de asinh \(math\)](#).

2308 Palabra clave 'assert' de Python

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Palabra clave 'assert' de Python](#).

2150 atan (math) — Devuelve la tangente inversa de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de atan \(math\)](#).

2151 atan2 (math) — Devuelve el ángulo principal del punto (x, y), teniendo en cuenta los signos de ambos argumentos para...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de atan2 \(math\)](#).

2144 atanh (math) — Devuelve la tangente hiperbólica inversa de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de atanh \(math\)](#).

2117 axis (matplotlib) — Establece el tamaño de la ventana de gráficos.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de axis \(matplotlib\)](#).

0859 Funciones en Python: 'B'

ID 859 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'B'

2124 barplot (matplotlib) — Crea un gráfico de barras correspondiente a la lista de datos proporcionada.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de barplot \(matplotlib\)](#).

1898 bin (builtins) — Convierte un número entero `x` en una cadena binaria con el prefijo `0b`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de bin \(builtins\)](#).

2054 black (graphic) — La constante `black` tiene un valor de 0, correspondiente al color negro en el modelo RGB888 de 24 bits,...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de black \(graphic\)](#).

2259 blit (hprime) — Copia todo el `gráfico_fuente` en el `gráfico_objetivo` en las coordenadas especificadas (`x`, `y`).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de blit \(hprime\)](#).

2275 blit_c (hpprime) — Copia todo el ``gráfico_fuente`` en el ``gráfico_objetivo`` en las coordenadas cartesianas especificadas...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de blit_c \(hpprime\)](#).

2051 blue (graphic) — La constante ``blue`` tiene un valor de 255, correspondiente al color azul en el modelo RGB888 de 24 bits,...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de blue \(graphic\)](#).

1757 bool, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de bool, clase \(builtins\)](#).

2120 boxplot (matplotlib) — Crea un diagrama de caja a partir de la lista de datos proporcionada.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de boxplot \(matplotlib\)](#).

2304 break (con bucle)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de break \(con bucle\)](#).

1786 bytearray, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de bytearray, clase \(builtins\)](#).

1787 append (builtins.bytearray) — Agrega un solo byte `b` (un entero entre 0 y 255) al final del bytearray.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de append \(builtins.bytearray\)](#).

1788 extend (builtins.bytearray) — Extiende el bytearray al agregar el contenido de ``input``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de extend \(builtins.bytearray\)](#).

2190 byteorder (sys) — El orden de bytes del sistema ("little" o "big").

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de byteorder \(sys\)](#).

1758 bytes, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de bytes, clase \(builtins\)](#).

1778 center (builtins.bytes) — Devuelve una copia del objeto centrado en una secuencia de longitud ``width``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de center \(builtins.bytes\)](#).

1775 count (builtins.bytes) — Devuelve el número de ocurrencias no superpuestas de la subsecuencia ``sub`` en el rango `[start, end]`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de count \(builtins.bytes\)](#).

1759 decode (builtins.bytes) — Devuelve una cadena decodificada de los bytes dados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de decode \(builtins.bytes\)](#).

1769 endswith (builtins.bytes) — Devuelve True si los datos binarios terminan con el sufijo especificado, de lo contrario devuelve False.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de endswith \(builtins.bytes\)](#).

1760 find (builtins.bytes) — Devuelve el índice más bajo en los datos donde se encuentra la subsecuencia `sub`, de modo que `sub`...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de find \(builtins.bytes\)](#).

1773 format (builtins.bytes) — Realiza una operación de formateo en un objeto bytes.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de format \(builtins.bytes\)](#).

1762 index (builtins.bytes) — Similar a find(), pero lanza ValueError cuando no se encuentra la subsecuencia.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de index \(builtins.bytes\)](#).

1782 isalpha (builtins.bytes) — Devuelve True si todos los bytes en la secuencia son caracteres ASCII alfabéticos y la secuencia no está...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isalpha \(builtins.bytes\)](#).

1783 isdigit (builtins.bytes) — Devuelve True si todos los bytes en la secuencia son dígitos decimales ASCII y la secuencia no está...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isdigit \(builtins.bytes\)](#).

1785 islower (builtins.bytes) — Devuelve True si hay al menos un carácter ASCII en minúscula en la secuencia y ningún carácter ASCII en...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de islower \(builtins.bytes\)](#).

1781 isspace (builtins.bytes) — Devuelve True si todos los bytes en la secuencia son espacios en blanco ASCII y la secuencia no está...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isspace \(builtins.bytes\)](#).

1784 isupper (builtins.bytes) — Devuelve True si hay al menos un carácter alfabético ASCII en mayúsculas en la secuencia y ningún...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isupper \(builtins.bytes\)](#).

1764 join (builtins.bytes) — Devuelve un objeto bytes o bytearray que es la concatenación de las secuencias de datos binarios en...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de join \(builtins.bytes\)](#).

1779 lower (builtins.bytes) — Devuelve una copia de la secuencia con todos los caracteres ASCII en mayúsculas convertidos a su...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de lower \(builtins.bytes\)](#).

1771 lstrip (builtins.bytes) — Devuelve una copia de la secuencia con los bytes iniciales especificados eliminados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de lstrip \(builtins.bytes\)](#).

1776 partition (builtins.bytes) — Divide la secuencia en la primera aparición de `sep`, y devuelve una tupla de 3 elementos que contiene...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de partition \(builtins.bytes\)](#).

1774 replace (builtins.bytes) — Devuelve una copia de la secuencia con todas las ocurrencias de la subsecuencia `old` reemplazadas por...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de replace \(builtins.bytes\)](#).

1761 rfind (builtins.bytes) — Devuelve el índice más alto en la secuencia donde se encuentra la subsecuencia sub, de modo que sub esté...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rfind \(builtins.bytes\)](#).

1763 rindex (builtins.bytes) — Similar a rfind() pero lanza un ValueError cuando no se encuentra la subsecuencia sub.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rindex \(builtins.bytes\)](#).

1777 rpartition (builtins.bytes) — Divide la secuencia en la última aparición de `sep` y devuelve una tupla de 3 elementos que contiene la...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rpartition \(builtins.bytes\)](#).

1767 rsplit (builtins.bytes) — Divide la secuencia binaria en subsecuencias del mismo tipo, usando sep como la cadena delimitadora.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rsplit \(builtins.bytes\)](#).

1772rstrip (builtins.bytes) — Devuelve una copia de la secuencia con los bytes finales especificados eliminados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rstrip \(builtins.bytes\)](#).

1765 split (builtins.bytes) — Divide la secuencia binaria en subsecuencias del mismo tipo, usando `sep` como la cadena delimitadora.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de split \(builtins.bytes\)](#).

1766 splitlines (builtins.bytes) — Devuelve una lista de las líneas en la secuencia binaria, dividiéndolas en los límites de línea ASCII.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de splitlines \(builtins.bytes\)](#).

1768 startswith (builtins.bytes) — Devuelve True si los datos binarios comienzan con el prefijo especificado, de lo contrario devuelve...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de startswith \(builtins.bytes\)](#).

1770 strip (builtins.bytes) — Devuelve una copia de la secuencia con los bytes iniciales y finales especificados eliminados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de strip \(builtins.bytes\)](#).

1780 upper (builtins.bytes) — Devuelve una copia de la secuencia con todos los caracteres ASCII en minúsculas convertidos a su...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de upper \(builtins.bytes\)](#).

2015 BytesIO, clase (uio)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de BytesIO, clase \(uio\)](#).

2016 read (uio.BytesIO) — El método read de la clase BytesIO se utiliza para leer datos del búfer de bytes en memoria.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de read \(uio.BytesIO\)](#).

2017 readinto (uio.BytesIO) — Lee bytes en un objeto de bytes preasignado y escribible `buffer`, y devuelve el número de bytes leídos.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readinto \(uio.BytesIO\)](#).

2018 readline (uio.BytesIO) — Lee una línea del contenido del objeto BytesIO, comenzando desde la posición actual del puntero y...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readline \(uio.BytesIO\)](#).

2019 write (uio.BytesIO) — Escribe los datos dados `data` (bytes, bytearray u objeto de cadena) en el búfer de bytes en memoria del...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de write \(uio.BytesIO\)](#).

2020 seek (uio.BytesIO) — El método seek ajusta el puntero a una ubicación específica dentro del búfer BytesIO, dependiendo de los...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de seek \(uio.BytesIO\)](#).

2021 flush (uio.BytesIO) — El método flush en la clase BytesIO está diseñado para ser compatible con las expectativas de una...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de flush \(uio.BytesIO\)](#).

2022 close (uio.BytesIO) — El método close de la clase BytesIO se utiliza para indicar que el objeto ya no es necesario y para...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de close \(uio.BytesIO\)](#).

2023 getvalue (uio.BytesIO) — Recupera todo el contenido del búfer de un objeto BytesIO.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de getvalue \(uio.BytesIO\).](#)

0860 Funciones en Python: 'C'

ID 860 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'C'

2042 calcsiz (ustruct) — Devuelve el número de bytes requeridos para almacenar una estructura definida por el parámetro...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de calcsiz \(ustruct\).](#)

1899 callable (builtins) — Devuelve True si el objeto `obj` parece ser llamable, False si no.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de callable \(builtins\).](#)

2128 caseval (cas) — Acepta una expresión como una cadena.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de caseval \(cas\).](#)

2152 ceil (math) — Devuelve un entero, siendo x redondeado hacia el infinito positivo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ceil \(math\).](#)

1778 center (builtins.bytes) — Devuelve una copia del objeto centrado en una secuencia de longitud `width`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de center \(builtins.bytes\).](#)

1877 center (builtins.str) — Devuelve una copia de la cadena centrada en una nueva cadena de longitud `width`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de center \(builtins.str\).](#)

2077 char (arit) — Devuelve el carácter que representa el unicode especificado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de char \(arit\).](#)

2250 choice (urandom) — Devuelve un elemento aleatorio de la secuencia no vacía `seq`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de choice \(urandom\).](#)

1901 chr (builtins) — Devuelve la cadena que representa un carácter cuyo punto de código Unicode es el entero `i`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de chr \(builtins\).](#)

2255 circle (hprime) — Dibuja un círculo en el `gráfico` especificado, centrado en `(x`, `y`), con un `radio` y un `color`...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de circle \(hprime\).](#)

2271 circle_c (hprime) — Dibuja un círculo en el `gráfico` especificado, centrado en las coordenadas cartesianas (`x`, `y`), con...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de circle_c \(hprime\)](#).

2313 class (bloque class)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de class \(bloque class\)](#).

1891 classmethod (builtins) — Transforma un método en un método de clase.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de classmethod \(builtins\)](#).

2305 continue (con bucle)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de continue \(con bucle\)](#).

1791 clear (builtins.dict) — Elimina todos los elementos del objeto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de clear \(builtins.dict\)](#).

1819 clear (builtins.list) — Elimina todos los elementos de la lista, dejándola vacía.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de clear \(builtins.list\)](#).

1840 clear (builtins.set) — Elimina todos los elementos del conjunto, dejándolo vacío.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de clear \(builtins.set\)](#).

2030 clear (ucollections.OrderedDict) — Elimina todos los elementos del objeto

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de clear \(ucollections.OrderedDict\)](#).

2068 clear_screen (graphic) — Limpia la pantalla llenándola con el `color` especificado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de clear_screen \(graphic\)](#).

2116 clf (matplotlib) — Elimina todos los objetos geométricos de la vista de gráfico.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de clf \(matplotlib\)](#).

1994 close (uio.FileIO) — Cierra el archivo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de close \(uio.FileIO\)](#).

2005 close (uio.TextIOWrapper) — Cierra el archivo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de close \(uio.TextIOWrapper\)](#).

2013 close (uio.StringIO) — El método `close` en la clase `StringIO` se utiliza para indicar que el objeto ya no es necesario y para...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de close \(uio.StringIO\)](#).

2022 close (uio.BytesIO) — El método `close` de la clase `BytesIO` se utiliza para indicar que el objeto ya no es necesario y para...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de close \(uio.BytesIO\)](#).

2200 collect (gc) — Inicia un ciclo de recolección de basura.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de collect \(gc\)](#).

1900 compile (builtins) — Compila el código fuente en un objeto de código.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de compile \(builtins\)](#).

2232 compile (ure) — Esta función es parte del módulo `ure`, que actualmente no es compatible con el HP Prime.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de compile \(ure\)](#).

1789 complex, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de complex, clase \(builtins\)](#).

2112 conj (linalg) — Devuelve una matriz que contiene el conjugado complejo de `x`, elemento por elemento.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de conj \(linalg\)](#).

1969 const (micropython) — Se utiliza para declarar que la `expresión` es una constante para que el compilador pueda optimizarla.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de const \(micropython\)](#).

1792 copy (builtins.dict) — Devuelve una copia superficial del objeto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de copy \(builtins.dict\)](#).

1806 copy (builtins.frozens) — Devuelve una copia superficial del `frozenset` (dado que el `frozenset` es inmutable, esto devolverá el...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de copy \(builtins.frozens\)](#).

1820 copy (builtins.list) — Devuelve una copia superficial de la lista.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de copy \(builtins.list\)](#).

1841 copy (builtins.set) — Devuelve una copia superficial del conjunto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de copy \(builtins.set\)](#).

2031 copy (ucollections.OrderedDict) — Devuelve una copia superficial del objeto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de copy \(ucollections.OrderedDict\)](#).

2153 copysign (math) — Devuelve x con el signo de y.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de copysign \(math\)](#).

2145 cos (math) — Devuelve el coseno de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de cos \(math\)](#).

2182 cos (cmath) — Devuelve el coseno de un número complejo `z`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de cos \(cmath\)](#).

2139 cosh (math) — Devuelve el coseno hiperbólico de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de cosh \(math\)](#).

1775 count (builtins.bytes) — Devuelve el número de ocurrencias no superpuestas de la subsecuencia `sub` en el rango `[start, end]`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de count \(builtins.bytes\)](#).

1821 count (builtins.list) — Devuelve el número de ocurrencias del elemento x en la lista.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de count \(builtins.list\)](#).

1874 count (builtins.str) — Devuelve el número de ocurrencias no superpuestas de la subcadena `sub` en el rango `[start, end]`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de count \(builtins.str\)](#).

1887 count (builtins.tuple) — Devuelve el número de ocurrencias del elemento value en la tupla.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de count \(builtins.tuple\)](#).

2087 cross (linalg) — Devuelve el producto cruzado de x e y, que son vectores de tamaño 3.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de cross \(linalg\)](#).

2048 cyan (graphic) — La constante ``cyan`` tiene un valor de 65535, correspondiente al color cian en el modelo RGB888 de 24...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de cyan \(graphic\)](#).

0861 Funciones en Python: 'D'

ID 861 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'D'

2235 DEBUG (ure) — Esta función es parte del módulo ure, que actualmente no es compatible con el HP Prime.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de DEBUG \(ure\)](#).

1759 decode (builtins.bytes) — Devuelve una cadena decodificada de los bytes dados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de decode \(builtins.bytes\)](#).

2293 def (bloque de función)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de def \(bloque de función\)](#).

2166 degrees (math) — Devuelve los radianes x convertidos a grados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de degrees \(math\)](#).

1902 delattr (builtins) — Este es un pariente de setattr().

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de delattr \(builtins\)](#).

1835 deleter (builtins.property) — El método deleter asigna o modifica la función de eliminación para una propiedad.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de deleter \(builtins.property\)](#).

2025 deque, clase (collections)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de deque, clase \(collections\)](#).

2026 append (collections) — Agrega el elemento ``x`` al extremo derecho del objeto deque.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de append \(collections\)](#).

2027 popleft (collections) — Elimina y devuelve un elemento del lado izquierdo del deque.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de popleft \(collections\)](#).

2099 det (linalg) — Devuelve el determinante de una matriz cuadrada.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de det \(linalg\)](#).

1790 dict, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de dict, clase \(builtins\)](#).

1791 clear (builtins.dict) — Elimina todos los elementos del objeto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de clear \(builtins.dict\)](#).

1792 copy (builtins.dict) — Devuelve una copia superficial del objeto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de copy \(builtins.dict\)](#).

1793 fromkeys (builtins.dict) — Crea un nuevo diccionario con claves del iterable y valores establecidos en value.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de fromkeys \(builtins.dict\)](#).

1794 get (builtins.dict) — Devuelve el valor de `key` si `key` está en el diccionario, de lo contrario devuelve `default`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de get \(builtins.dict\)](#).

1795 items (builtins.dict) — Devuelve una nueva vista de los elementos del diccionario (pares (clave, valor)).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de items \(builtins.dict\)](#).

1796 keys (builtins.dict) — Devuelve una nueva vista de las claves del diccionario

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de keys \(builtins.dict\)](#).

1797 pop (builtins.dict) — Si la clave está en el diccionario, se elimina y se devuelve su valor; de lo contrario, se devuelve el...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pop \(builtins.dict\)](#).

1798 popitem (builtins.dict) — Elimina y devuelve un par (clave, valor) del diccionario.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de popitem \(builtins.dict\)](#).

1799 setdefault (builtins.dict) — Si la clave está en el diccionario, devuelve su valor.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de setdefault \(builtins.dict\)](#).

1800 update (builtins.dict) — Actualiza el diccionario con los pares clave/valor de otro, sobrescribiendo las claves existentes.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de update \(builtins.dict\)](#).

1801 values (builtins.dict) — Devuelve una nueva vista de los valores del diccionario.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de values \(builtins.dict\)](#).

1807 difference (builtins.frozens) — Devuelve un nuevo frozenset que contiene elementos que están en este frozenset pero no en los otros.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de difference \(builtins.frozens\)](#).

1843 difference (builtins.set) — Devuelve un nuevo conjunto que contiene elementos que están presentes en el conjunto original pero no en...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de difference \(builtins.set\)](#).

1844 difference_update (builtins.set) — Elimina los elementos del conjunto original que también están presentes en cualquiera de los otros...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de difference_update \(builtins.set\)](#).

2244 digest (uhashlib.sha256) — Devuelve el hash de todos los datos pasados a través del hash, como un objeto de bytes.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de digest \(uhashlib.sha256\)](#).

2262 dimgrob (hpprime) — Redimensiona `gráfico` a las dimensiones especificadas `ancho` y `alto`, y lo llena con el `color`...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de dimgrob \(hpprime\)](#).

2278 dimgrob_c (hpprime) — Redimensiona `gráfico` a las dimensiones especificadas `ancho` y `alto` medidas en unidades cartesianas,...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de dimgrob_c \(hpprime\)](#).

1903 dir (builtins) — Sin argumentos, devuelve la lista de nombres en el ámbito local actual.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de dir \(builtins\)](#).

2201 disable (gc) — Desactiva la recolección automática de basura.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de disable \(gc\)](#).

1842 discard (builtins.set) — Elimina el elemento especificado `e` del conjunto si está presente.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de discard \(builtins.set\)](#).

1904 divmod (builtins) — Toma dos números (no complejos) `a` y `b` como argumentos y devuelve un par de números que consiste en...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de divmod \(builtins\)](#).

2086 dot (linalg) — Devuelve el producto punto de a y b.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de dot \(linalg\)](#).

2063 draw_arc (graphic) — Dibuja un arco cerrado en la pantalla, definido por un segmento elíptico con centro en (`x`, `y`).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de draw_arc \(graphic\)](#).

2061 draw_circle (graphic) — Dibuja un círculo en la pantalla, definido por su centro en (`x`, `y`) y su `radio`, utilizando el...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de draw_circle \(graphic\)](#).

2064 draw_filled_arc (graphic) — Dibuja un sector relleno de una elipse en la pantalla, definido por un segmento de arco con centro en...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de draw_filled_arc \(graphic\)](#).

2062 draw_filled_circle (graphic) — Dibuja un círculo relleno en la pantalla, definido por su centro en (`x`, `y`) y su `radio`, utilizando...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de draw_filled_circle \(graphic\)](#).

2059 draw_filled_polygon (graphic) — Dibuja un polígono relleno definido por una lista o tupla de coordenadas (`puntos`) utilizando el color...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de draw_filled_polygon \(graphic\)](#).

2060 draw_filled_rectangle (graphic) — Dibuja un rectángulo relleno en la pantalla, definido por su esquina superior izquierda en (`x`, `y`),...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de draw_filled_rectangle \(graphic\)](#).

2057 draw_line (graphic) — Dibuja una línea entre los puntos (`x1`, `y1`) y (`x2`, `y2`) en la pantalla utilizando el color especificado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de draw_line \(graphic\)](#).

2058 draw_polygon (graphic) — Dibuja un polígono definido por una lista o tupla de coordenadas (`puntos`) utilizando el color...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de draw_polygon \(graphic\)](#).

2066 draw_string (graphic) — Dibuja un texto en la pantalla en las coordenadas especificadas (`x`, `y`).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de draw_string \(graphic\)](#).

0862 Funciones en Python: 'E'

ID 862 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'E'

2131 e (math) — Devuelve el valor de la constante e utilizada por el módulo matemático (2.71828182845905).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de e \(math\)](#).

2172 e (cmath) — Devuelve el valor de la constante e utilizada por el módulo cmath (2.71828182845905).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de e \(cmath\)](#).

2094 eig (linalg) — Devuelve los vectores propios y los valores propios de una matriz cuadrada x como una tupla...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de eig \(linalg\)](#).

2093 eigenvects (linalg) — Devuelve los vectores propios de una matriz cuadrada.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de eigenvects \(linalg\)](#).

2299 elif (bloque if-elif-else)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de elif \(bloque if-elif-else\)](#).

1893 Ellipsis en Python

ID 1893 · título recuperado de la página española

Sintaxis

Ellipsis
...

Ellipsis aparece en el código como la palabra clave 'Ellipsis' o en forma de tres puntos (...). La forma de (...) se utiliza comúnmente como un marcador de posición en el código que planea completar más tarde. Esto es particularmente útil en las definiciones de clases y funciones que se implementarán en el futuro. Dado que Ellipsis es un objeto, se puede asignar a una variable o convertirse en el valor predeterminado de un parámetro de función.

Ejemplo

```
def fn_test(val = ...):
    if val is Ellipsis:
        print('Valor predeterminado')
    else:
        print('Valor: ' + str(val))
    ... # marcador de posición
```

2298 else (bloque if-else)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de else \(bloque if-else\)](#).

2302 else (con bucle for)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de else \(con bucle for\)](#).

2202 enable (gc) — Reactiva la recolección automática de basura si ha sido deshabilitada previamente usando el método...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de enable \(gc\)](#).

1858 encode (builtins.str) — Devuelve una versión codificada de la cadena como un objeto de bytes.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de encode \(builtins.str\)](#).

1769 endswith (builtins.bytes) — Devuelve True si los datos binarios terminan con el sufijo especificado, de lo contrario devuelve False.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de endswith \(builtins.bytes\)](#).

1868 endswith (builtins.str) — Devuelve True si la cadena termina con el sufijo especificado ``suffix``, de lo contrario devuelve False.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de endswith \(builtins.str\)](#).

1802 enumerate (builtins) — Devuelve un objeto enumerado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de enumerate \(builtins\)](#).

2167 erf (math) — Devuelve la función de error de x .

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de erf \(math\)](#).

2168 erfc (math) — Devuelve la función de error complementaria de x .

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de erfc \(math\)](#).

2079 euler (arit) — Devuelve la función ϕ de Euler evaluada en un entero ``n``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de euler \(arit\)](#).

1905 eval (builtins) — ``expression``: una cadena que es la expresión de Python a evaluar.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de eval \(builtins\)](#).

2289 eval (hprime) — Úa la cadena como una línea de comando PPL.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de eval \(hprime\)](#).

2197 exc_info (sys) — Devuelve información sobre la excepción que se está manejando actualmente.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de exc_info \(sys\)](#).

1906 exec (builtins) — Descripción: Esta función permite la ejecución dinámica de código Python.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de exec \(builtins\)](#).

2192 exit (sys) — Termina el programa con un estado de salida opcional.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de exit \(sys\)](#).

2134 exp (math) — Devuelve e elevado a la potencia de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de exp \(math\)](#).

2177 exp (cmath) — Devuelve la exponencial de un número complejo `z`, equivalente a e elevado a la potencia de `z`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de exp \(cmath\)](#).

2135 expm1 (math) — Devuelve $\exp(x) - 1$.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de expm1 \(math\)](#).

1788 extend (builtins bytearray) — Extiende el bytearray al agregar el contenido de `input`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de extend \(builtins bytearray\)](#).

1822 extend (builtins list) — Extiende la lista al agregar todos los elementos del iterable dado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de extend \(builtins list\)](#).

1981 extend (array.array) — Extiende el arreglo agregando todos los elementos del iterable proporcionado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de extend \(array.array\)](#).

0863 Funciones en Python: 'F'

ID 863 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'F'

2154 fabs (math) — Devuelve el valor absoluto de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de fabs \(math\)](#).

2097 fft (linalg) — Devuelve la transformada discreta de Fourier de un vector.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de fft \(linalg\)](#).

1984 FileIO, clase (uio)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de FileIO, clase \(uio\)](#).

1985 fileno (uio.FileIO) — Devuelve el descriptor de archivo asociado con un objeto de E/S.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de fileno \(uio.FileIO\)](#).

1986 read (uio.FileIO) — El método read se utiliza para leer un número especificado de bytes (`size`) de un archivo abierto (o...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de read \(uio.FileIO\)](#).

1987 readinto (uio.FileIO) — Lee bytes en un objeto de tipo bytes preasignado y escribible, especificado por el búfer ``buffer``...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readinto \(uio.FileIO\)](#).

1988 readline (uio.FileIO) — El método `readline` lee una línea del archivo, comenzando desde la posición actual del puntero del...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readline \(uio.FileIO\)](#).

1989 readlines (uio.FileIO) — El método `readlines` lee múltiples líneas del archivo, comenzando desde la posición actual del puntero...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readlines \(uio.FileIO\)](#).

1990 write (uio.FileIO) — El método `write` escribe los datos especificados por el parámetro ``data`` (bytes, bytearray o string) en...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de write \(uio.FileIO\)](#).

1991 seek (uio.FileIO) — El método `seek` ajusta el puntero del archivo a una ubicación específica dentro del archivo, dependiendo...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de seek \(uio.FileIO\)](#).

1992 tell (uio.FileIO) — Devuelve la posición actual del puntero de archivo dentro del archivo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de tell \(uio.FileIO\)](#).

1993 flush (uio.FileIO) — El método `flush` se utiliza para limpiar cualquier búfer interno del archivo y garantizar que todos los...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de flush \(uio.FileIO\)](#).

1994 close (uio.FileIO) — Cierra el archivo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de close \(uio.FileIO\)](#).

1985 fileno (uio.FileIO) — Devuelve el descriptor de archivo asociado con un objeto de E/S.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de fileno \(uio.FileIO\)](#).

1996 fileno (uio.TextIOWrapper) — Devuelve el descriptor de archivo asociado con un objeto `TextIOWrapper`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de fileno \(uio.TextIOWrapper\)](#).

2258 fillpoly (hprime) — Dibuja un polígono relleno en el gráfico especificado por ``graphic`` utilizando coordenadas de píxeles.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de fillpoly \(hprime\)](#).

2274 fillpoly_c (hprime) — Dibuja un polígono relleno en el gráfico especificado por ``graphic`` utilizando coordenadas cartesianas.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de fillpoly_c \(hprime\)](#).

2268 fillrect (hprime) — Dibuja un rectángulo en el gráfico especificado en `(`x`, `y`)` con las dimensiones especificadas...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de fillrect \(hprime\)](#).

2284 fillrect_c (hprime) — Dibuja un rectángulo en el gráfico especificado en las coordenadas cartesianas `(`x`, `y`)` con las...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de fillrect_c \(hprime\)](#).

1803 filter (builtins) — Construye un iterador a partir de aquellos elementos de ``iterable`` para los cuales ``function`` devuelve...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de filter \(builtins\)](#).

1760 find (builtins.bytes) — Devuelve el índice más bajo en los datos donde se encuentra la subsecuencia ``sub``, de modo que ``sub``...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de find \(builtins.bytes\)](#).

1859 find (builtins.str) — Devuelve el índice más bajo en la cadena donde se encuentra la subcadena ``sub``, de modo que ``sub`` esté...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de find \(builtins.str\)](#).

1804 float, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de float, clase \(builtins\)](#).

2155 floor (math) — Devuelve un entero, siendo `x` redondeado hacia el infinito negativo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de floor \(math\)](#).

1993 flush (uio.FileIO) — El método `flush` se utiliza para limpiar cualquier búfer interno del archivo y garantizar que todos los...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de flush \(uio.FileIO\)](#).

2004 flush (uio.TextIOWrapper) — El método `flush` se utiliza para limpiar cualquier búfer interno del archivo y garantizar que todos los...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de flush \(uio.TextIOWrapper\)](#).

2012 flush (uio.StringIO) — El método flush en la clase StringIO está diseñado para ser compatible con las expectativas de una...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de flush \(uio.StringIO\)](#).

2021 flush (uio.BytesIO) — El método flush en la clase BytesIO está diseñado para ser compatible con las expectativas de una...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de flush \(uio.BytesIO\)](#).

2157 fmod (math) — Devuelve el resto de x/y.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de fmod \(math\)](#).

2300 for (bucle de secuencia)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de for \(bucle de secuencia\)](#).

2301 for (bucle de rango)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de for \(bucle de rango\)](#).

1773 format (builtins.bytes) — Realiza una operación de formateo en un objeto bytes.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de format \(builtins.bytes\)](#).

1872 format (builtins.str) — Realiza una operación de formateo de cadenas.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de format \(builtins.str\)](#).

2158 frexp (math) — Descompone un número de punto flotante en su mantisa y exponente.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de frexp \(math\)](#).

1793 fromkeys (builtins.dict) — Crea un nuevo diccionario con claves del iterable y valores establecidos en value.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de fromkeys \(builtins.dict\)](#).

2032 Método de clase fromkeys de Python

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Método de clase fromkeys de Python](#).

1815 from_bytes (builtins.int) — Devuelve un entero sin signo representado por el arreglo de bytes dado en el orden de bytes especificado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de from_bytes \(builtins.int\)](#).

1805 frozenset, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de frozenset, clase \(builtins\)](#).

1806 copy (builtins.frozens) — Devuelve una copia superficial del frozenset (dado que el frozenset es inmutable, esto devolverá el...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de copy \(builtins.frozens\)](#).

1807 difference (builtins.frozens) — Devuelve un nuevo frozenset que contiene elementos que están en este frozenset pero no en los otros.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de difference \(builtins.frozens\)](#).

1808 intersection (builtins.frozens) — Devuelve un nuevo frozenset que contiene los elementos que son comunes entre este frozenset y todos los...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de intersection \(builtins.frozens\)](#).

1809 isdisjoint (builtins.frozens) — Devuelve True si el frozenset no tiene elementos en común con otro, de lo contrario devuelve False.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isdisjoint \(builtins.frozens\)](#).

1810 issubset (builtins.frozens) — Verifica si cada elemento en el conjunto está en el otro.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de issubset \(builtins.frozens\)](#).

1811 issuperset (builtins.frozens) — Devuelve True si este frozenset contiene todos los elementos de otro.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de issuperset \(builtins.frozens\)](#).

1812 symmetric_difference (builtins.frozens) — Devuelve un nuevo frozenset que contiene elementos que están en este frozenset o en otro, pero no en...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de symmetric_difference \(builtins.frozens\)](#).

1813 union (builtins.frozens) — Devuelve un nuevo frozenset con elementos de este frozenset y de todos los demás.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de union \(builtins.frozens\)](#).

0864 Funciones en Python: 'G'

ID 864 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'G'

2169 gamma (math) — Devuelve la función gamma de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de gamma \(math\)](#).

2074 gcd (arit) — Devuelve el máximo común divisor de los enteros ``a`` y ``b``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de gcd \(arit\)](#).

1794 get (builtins.dict) — Devuelve el valor de ``key`` si ``key`` está en el diccionario, de lo contrario devuelve ``default``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de get \(builtins.dict\)](#).

2033 get (ucollections.OrderedDict) — Devuelve el valor de la clave si la clave está en el diccionario, de lo contrario devuelve el valor...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de get \(ucollections.OrderedDict\)](#).

1907 getattr (builtins) — Devuelve el valor del atributo nombrado del objeto ``obj``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de getattr \(builtins\)](#).

2263 getpix (hprime) — Recupera el color del píxel del ``gráfico`` especificado en las coordenadas `(`x`,`y`)` y lo devuelve como...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de getpix \(hprime\)](#).

2279 getpix_c (hprime) — Recupera el color del píxel del ``gráfico`` especificado en las coordenadas cartesianas `(`x`,`y`)` y lo...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de getpix_c \(hprime\)](#).

2246 getrandbits (urandom) — Devuelve un entero aleatorio que consiste en un máximo de ``k`` bits, donde ``k`` varía de 0 a 32.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de getrandbits \(urandom\)](#).

1833 getter (builtins.property) — El método getter asigna o modifica la función getter para una propiedad.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de getter \(builtins.property\)](#).

2014 getvalue (uio.StringIO) — Recupera todo el contenido del búfer de un objeto StringIO.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de getvalue \(uio.StringIO\)](#).

2023 getvalue (uio.BytesIO) — Recupera todo el contenido del búfer de un objeto BytesIO.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de getvalue \(uio.BytesIO\)](#).

2290 get_cartesian (hprime) — Devuelve una lista en la forma de `[xmin, xmax, ymin, ymax]` que contiene los rangos de los ejes x e y...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de get_cartesian \(hprime\)](#).

2129 get_key (cas) — Espera la próxima pulsación de tecla y devuelve un valor entero de 0 a 50, que identifica teclas...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de get_key \(cas\)](#).

2065 get_pixel (graphic) — Recupera el color del píxel ubicado en las coordenadas especificadas (x, y).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de get_pixel \(graphic\)](#).

2310 Palabra clave 'global' de Python

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Palabra clave 'global' de Python](#).

1909 globals (builtins) — Devuelve un diccionario que representa la tabla de símbolos global actual.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de globals \(builtins\)](#).

2053 green (graphic) — La constante ``green`` tiene un valor de 65280, correspondiente al color verde en el modelo RGB888 de 24...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de green \(graphic\)](#).

2121 grid (matplotlib) — Con 0 como argumento, hace que los ejes del gráfico no sean visibles ('axes = 0').

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de grid \(matplotlib\)](#).

2265 grobh (hprime) — Devuelve la altura en píxeles del gráfico especificado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de grobh \(hprime\)](#).

2281 grobh_c (hprime) — Devuelve la altura en unidades cartesianas del gráfico especificado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de grobh_c \(hprime\)](#).

2264 grobw (hprime) — Devuelve el ancho en píxeles del gráfico especificado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de grobw \(hprime\)](#).

2280 grobw_c (hprime) — Devuelve el ancho en unidades cartesianas del gráfico especificado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de grobw_c \(hprime\)](#).

0865 Funciones en Python: 'H'

ID 865 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'H'

1910 hasattr (builtins) — Los argumentos son un objeto y una cadena.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de hasattr \(builtins\).](#)

1911 hash (builtins) — Devuelve el valor hash del objeto (si tiene uno).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de hash \(builtins\).](#)

1975 heap_lock (micropython) — Bloquea el montón.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de heap_lock \(micropython\).](#)

1976 heap_unlock (micropython) — Desbloquea el montón.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de heap_unlock \(micropython\).](#)

1912 help (builtins) — Invoca el sistema de ayuda integrado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de help \(builtins\).](#)

1913 hex (builtins) — Convierte un número entero en una cadena hexadecimal en minúsculas precedida por "0x".

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de hex \(builtins\).](#)

2123 histogram (matplotlib) — Crea un histograma correspondiente a la lista de datos proporcionada.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de histogram \(matplotlib\).](#)

0866 Funciones en Python: 'I'

ID 866 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'I'

1914 id (builtins) — Devuelve la 'identidad' de un objeto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de id \(builtins\).](#)

2103 identity (linalg) — Devuelve una matriz identidad cuadrada.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de identity \(linalg\).](#)

2076 iegcd (arit) — Devuelve el máximo común divisor extendido para los enteros ``a`` y ``b``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de iegcd \(arit\).](#)

2297 if (bloque if)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de if \(bloque if\).](#)

2073 ifactor (arit) — Devuelve la factorización de un entero `n` como un producto de potencias de primos.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ifactor \(arit\)](#).

2098 ifft (linalg) — Devuelve la transformada discreta de Fourier inversa de un vector.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ifft \(linalg\)](#).

2111 imag (linalg) — Devuelve un vector o matriz que contiene la parte imaginaria de `x`, elemento por elemento.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de imag \(linalg\)](#).

2188 implementation (sys) — Objeto con información sobre la implementación actual de Python.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de implementation \(sys\)](#).

1762 index (builtins.bytes) — Similar a `find()`, pero lanza `ValueError` cuando no se encuentra la subsecuencia.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de index \(builtins.bytes\)](#).

1823 index (builtins.list) — Devuelve el índice de la primera ocurrencia del elemento `x` en la lista.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de index \(builtins.list\)](#).

1861 index (builtins.str) — Funciona como `find()`, pero genera un `ValueError` cuando no se encuentra la subcadena `sub`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de index \(builtins.str\)](#).

1888 index (builtins.tuple) — Devuelve el índice de la primera ocurrencia del elemento `value` en la tupla.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de index \(builtins.tuple\)](#).

1915 input (builtins) — Si el argumento `mensaje` está presente, se escribe en la salida estándar sin un salto de línea final.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de input \(builtins\)](#).

1824 insert (builtins.list) — A el elemento `x` en el índice especificado `i`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de insert \(builtins.list\)](#).

1814 int, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de int, clase \(builtins\)](#).

1815 from_bytes (builtins.int) — Devuelve un entero sin signo representado por el arreglo de bytes dado en el orden de bytes especificado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de from_bytes \(builtins.int\)](#).

1816 to_bytes (builtins.int) — Devuelve un array de bytes que representa un entero utilizando el orden de bytes especificado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de to_bytes \(builtins.int\)](#).

1808 intersection (builtins.frozens) — Devuelve un nuevo frozenset que contiene los elementos que son comunes entre este frozenset y todos los...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de intersection \(builtins.frozens\)](#).

1845 intersection (builtins.set) — Devuelve un nuevo conjunto que contiene los elementos que son comunes tanto al conjunto original como al...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de intersection \(builtins.set\)](#).

1846 intersection_update (builtins.set) — Actualiza el conjunto original para mantener solo los elementos que son comunes tanto al conjunto...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de intersection_update \(builtins.set\)](#).

2095 inv (linalg) — Dada una matriz cuadrada no singular, devuelve una matriz que contiene la inversa de x, satisfaciendo...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de inv \(linalg\)](#).

2257 invert (hprime) — Invierte un rectángulo en el `gráfico` especificado en (`x`, `y`) con las dimensiones especificadas...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de invert \(hprime\)](#).

2273 invert_c (hprime) — Invierte un rectángulo en el `gráfico` especificado en las coordenadas cartesianas (`x`, `y`) con las...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de invert_c \(hprime\)](#).

1782 isalpha (builtins.bytes) — Devuelve True si todos los bytes en la secuencia son caracteres ASCII alfabéticos y la secuencia no está...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isalpha \(builtins.bytes\)](#).

1881 isalpha (builtins.str) — Devuelve True si todos los caracteres de la cadena son caracteres ASCII alfabéticos y la cadena no está...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isalpha \(builtins.str\)](#).

1783 isdigit (builtins.bytes) — Devuelve True si todos los bytes en la secuencia son dígitos decimales ASCII y la secuencia no está...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isdigit \(builtins.bytes\)](#).

1882 isdigit (builtins.str) — Devuelve True si todos los caracteres de la cadena son dígitos decimales ASCII y la cadena no está...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isdigit \(builtins.str\)](#).

1809 isdisjoint (builtins.frozens) — Devuelve True si el frozenset no tiene elementos en común con otro, de lo contrario devuelve False.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isdisjoint \(builtins.frozens\)](#).

1847 isdisjoint (builtins.set) — Devuelve True si el conjunto original no tiene elementos en común con el conjunto ``other``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isdisjoint \(builtins.set\)](#).

2203 isenabled (gc) — Devuelve el estado actual de la recolección automática de basura.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isenabled \(gc\)](#).

2161 isfinite (math) — Devuelve True si `x` es finito.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isfinite \(math\)](#).

2162 isinf (math) — Devuelve True si `x` es infinito.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isinf \(math\)](#).

1916 isinstance (builtins) — Devuelve True si el argumento `object` es una instancia del argumento `classinfo`, o de una subclase...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isinstance \(builtins\)](#).

1785 islower (builtins.bytes) — Devuelve True si hay al menos un carácter ASCII en minúscula en la secuencia y ningún carácter ASCII en...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de islower \(builtins.bytes\)](#).

1884 islower (builtins.str) — Devuelve True si hay al menos un carácter alfabético ASCII en minúscula en la cadena y no hay caracteres...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de islower \(builtins.str\)](#).

2163 isnan (math) — Devuelve True si ``x`` no es un número.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isnan \(math\)](#).

2070 isprime (arit) — Devuelve un booleano que indica si un entero ``n`` es primo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isprime \(arit\)](#).

1781 isspace (builtins.bytes) — Devuelve True si todos los bytes en la secuencia son espacios en blanco ASCII y la secuencia no está...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isspace \(builtins.bytes\)](#).

1880 isspace (builtins.str) — Devuelve True si todos los caracteres de la cadena son espacios en blanco ASCII y la cadena no está...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isspace \(builtins.str\)](#).

1917 issubclass (builtins) — Devuelve True si la clase es una subclase (directa, indirecta o virtual) de classinfo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de issubclass \(builtins\)](#).

1810 issubset (builtins.frozens) — Verifica si cada elemento en el conjunto está en el otro.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de issubset \(builtins.frozens\)](#).

1848 issubset (builtins.set) — Devuelve True si todos los elementos del conjunto original están contenidos en el conjunto `other`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de issubset \(builtins.set\)](#).

1811 issuperset (builtins.frozens) — Devuelve True si este frozenset contiene todos los elementos de otro.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de issuperset \(builtins.frozens\)](#).

1849 issuperset (builtins.set) — Devuelve True si el conjunto original contiene todos los elementos del conjunto `other`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de issuperset \(builtins.set\)](#).

1784 isupper (builtins.bytes) — Devuelve True si hay al menos un carácter alfabético ASCII en mayúsculas en la secuencia y ningún...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isupper \(builtins.bytes\)](#).

1883 isupper (builtins.str) — Devuelve True si hay al menos un carácter alfabético ASCII en mayúsculas en la cadena y no hay...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isupper \(builtins.str\)](#).

1795 items (builtins.dict) — Devuelve una nueva vista de los elementos del diccionario (pares (clave, valor)).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de items \(builtins.dict\)](#).

2034 items (collections.OrderedDict) — Devuelve una nueva vista de los elementos del diccionario (pares (clave, valor)).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de items \(collections.OrderedDict\)](#).

1918 iter (builtins) — Devuelve un objeto iterador.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de iter \(builtins\).](#)

0867 Funciones en Python: 'J'

ID 867 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'J'

1764 join (builtins.bytes) — Devuelve un objeto bytes o bytearray que es la concatenación de las secuencias de datos binarios en...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de join \(builtins.bytes\).](#)

1863 join (builtins.str) — Devuelve una cadena que es la concatenación de las cadenas en el iterable `iter`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de join \(builtins.str\).](#)

0868 Funciones en Python: 'K'

ID 868 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'K'

1977 kbd_intr (micropython) — Establece o borra el carácter de interrupción del teclado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de kbd_intr \(micropython\).](#)

2286 keyboard (hprime) — Devuelve un número que representa el estado de cada tecla en el teclado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de keyboard \(hprime\).](#)

1796 keys (builtins.dict) — Devuelve una nueva vista de las claves del diccionario

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de keys \(builtins.dict\).](#)

2035 keys (ucollections.OrderedDict) — Devuelve una nueva vista de las claves del diccionario

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de keys \(ucollections.OrderedDict\).](#)

0869 Funciones en Python: 'L'

ID 869 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'L'

2306 Expresión lambda de Python

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Expresión lambda de Python.](#)

2075 lcm (arit) — Devuelve el mínimo común múltiplo de los enteros ``a`` y ``b``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de lcm \(arit\)](#).

2159 ldexp (math) — Devuelve $x * (2^{**exp})$.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ldexp \(math\)](#).

1919 len (builtins) — Devuelve la longitud (el número de elementos) de un objeto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de len \(builtins\)](#).

2170 lgamma (math) — Devuelve el logaritmo natural de la función gamma de x .

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de lgamma \(math\)](#).

2266 line (hprime) — Dibuja una línea en el `gráfico` especificado desde `(`x1`, `y1`)` hasta `(`x2`, `y2`)` utilizando el...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de line \(hprime\)](#).

2125 linear_regression_plot (matplotlib) — Dibuja los puntos individuales así como la regresión lineal correspondiente.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de linear_regression_plot \(matplotlib\)](#).

2282 line_c (hprime) — Dibuja una línea en el `gráfico` especificado desde las coordenadas cartesianas `(`x1`, `y1`)` hasta...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de line_c \(hprime\)](#).

2085 linspace (linalg) — Devuelve un vector de n muestras espaciadas uniformemente calculadas en el intervalo $[a, b]$.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de linspace \(linalg\)](#).

1817 list, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de list, clase \(builtins\)](#).

1818 append (builtins.list) — Agrega el elemento x al final de la lista.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de append \(builtins.list\)](#).

1819 clear (builtins.list) — Elimina todos los elementos de la lista, dejándola vacía.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de clear \(builtins.list\)](#).

1820 copy (builtins.list) — Devuelve una copia superficial de la lista.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de copy \(builtins.list\)](#).

1821 count (builtins.list) — Devuelve el número de ocurrencias del elemento *x* en la lista.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de count \(builtins.list\)](#).

1822 extend (builtins.list) — Extiende la lista al agregar todos los elementos del iterable dado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de extend \(builtins.list\)](#).

1823 index (builtins.list) — Devuelve el índice de la primera ocurrencia del elemento *x* en la lista.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de index \(builtins.list\)](#).

1824 insert (builtins.list) — A el elemento *x* en el índice especificado *i*.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de insert \(builtins.list\)](#).

1825 pop (builtins.list) — Elimina y devuelve el elemento en el índice especificado *i* de la lista.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pop \(builtins.list\)](#).

1826 remove (builtins.list) — Elimina la primera ocurrencia del elemento *x* de la lista.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de remove \(builtins.list\)](#).

1827 reverse (builtins.list) — Invierte los elementos de la lista en su lugar, lo que significa que la lista se modifica directamente y...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de reverse \(builtins.list\)](#).

1828 sort (builtins.list) — Ordena la lista en orden ascendente por defecto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de sort \(builtins.list\)](#).

1920 locals (builtins) — Actualiza y devuelve un diccionario que representa la tabla de símbolos local actual.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de locals \(builtins\)](#).

2136 log (math) — Devuelve el logaritmo natural de *x*.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de log \(math\)](#).

2178 log (cmath) — Devuelve el logaritmo natural de un número complejo *z* para la base 'e'.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de log \(cmath\)](#).

2138 log10 (math) — Devuelve el logaritmo en base 10 de *x*.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de log10 \(math\)](#).

2179 log10 (cmath) — Devuelve el logaritmo en base 10 de un número complejo `z`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de log10 \(cmath\)](#).

2137 log2 (math) — Devuelve el logaritmo en base 2 de `x`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de log2 \(math\)](#).

1779 lower (builtins.bytes) — Devuelve una copia de la secuencia con todos los caracteres ASCII en mayúsculas convertidos a su...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de lower \(builtins.bytes\)](#).

1878 lower (builtins.str) — Devuelve una copia de la cadena con todos los caracteres ASCII en mayúsculas convertidos a sus...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de lower \(builtins.str\)](#).

1771 lstrip (builtins.bytes) — Devuelve una copia de la secuencia con los bytes iniciales especificados eliminados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de lstrip \(builtins.bytes\)](#).

1870 lstrip (builtins.str) — Devuelve una copia de la cadena con los caracteres iniciales especificados eliminados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de lstrip \(builtins.str\)](#).

0870 Funciones en Python: 'M'

ID 870 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'M'

2049 magenta (graphic) — La constante `magenta` tiene un valor de 16711935, correspondiente al color magenta en el modelo RGB888...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de magenta \(graphic\)](#).

1829 map (builtins) — (función, iterable, ...)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de map \(builtins\)](#).

2233 match (ure) — Esta función es parte del módulo `ure`, que actualmente no es compatible con el HP Prime.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de match \(ure\)](#).

2083 matrix (linalg) — Devuelve una matriz de (n, m) con elementos contruidos a partir de `f`, que puede ser una función o una...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de matrix \(linalg\)](#).

1921 max (builtins) — Devuelve el elemento más grande en un iterable ``i`` o el más grande de dos o más argumentos.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de max \(builtins\).](#)

2191 maxsize (sys) — Valor máximo que un tipo de entero nativo puede contener en la plataforma actual, o valor máximo...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de maxsize \(sys\).](#)

1830 memoryview, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de memoryview, clase \(builtins\).](#)

2205 mem_alloc (gc) — Devuelve el número de bytes de RAM de montón que están asignados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de mem_alloc \(gc\).](#)

2204 mem_free (gc) — Devuelve el número de bytes de RAM de montón disponibles, o -1 si esta cantidad no se conoce.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de mem_free \(gc\).](#)

1971 mem_info (micropython) — Imprime información sobre la memoria actualmente utilizada.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de mem_info \(micropython\).](#)

1922 min (builtins) — Devuelve el elemento más pequeño en un iterable ``i`` o el más pequeño de dos o más argumentos.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de min \(builtins\).](#)

2160 modf (math) — Devuelve una tupla de dos flotantes, que son las partes fraccionaria e integral de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de modf \(math\).](#)

2196 modules (sys) — Diccionario de módulos que se han cargado en el entorno de MicroPython.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de modules \(sys\).](#)

2287 mouse (hprime) — Devuelve una tupla de 2 tuplas que representan los 2 puntos de contacto del mouse.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de mouse \(hprime\).](#)

2089 mul (linalg) — Devuelve el producto de matrices o el producto matriz-escalar de A y B.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de mul \(linalg\).](#)

0871 Funciones en Python: 'N'

Lista de funciones y comandos en Python que comienzan con 'N'

2028 namedtuple, clase (collections)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de namedtuple, clase \(collections\)](#).

1923 next (builtins) — Devuelve el siguiente elemento del iterador `iter``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de next \(builtins\)](#).

2071 nextprime (arit) — Devuelve el siguiente número primo mayor que un entero `n``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de nextprime \(arit\)](#).

2311 Palabra clave 'nonlocal' de Python

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Palabra clave 'nonlocal' de Python](#).

2080 nprimes (arit) — Devuelve el número de primos menores o iguales a un entero `n``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de nprimes \(arit\)](#).

0872 Funciones en Python: 'O'

Lista de funciones y comandos en Python que comienzan con 'O'

1831 object, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de object, clase \(builtins\)](#).

1924 oct (builtins) — Convierte un número entero `x`` en una cadena octal precedida por `"0o"`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de oct \(builtins\)](#).

2102 ones (linalg) — Devuelve un vector (si `m=0`) o una matriz (si `m>0`) llena de unos.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ones \(linalg\)](#).

1932 open (builtins) — La función `open()` abre un archivo y devuelve el objeto de archivo de tipo `TextIOWrapper`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de open \(builtins\)](#).

1983 open (uio) — La función `open()` abre un archivo en modo texto y devuelve el objeto de archivo correspondiente...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de open \(uio\)](#).

1970 `opt_level` (micropython) — Si se proporciona un nivel, esta función establece el nivel de optimización para la compilación...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `opt\_level` \(micropython\)](#).

1925 `ord` (builtins) — Dada una cadena que representa un solo carácter Unicode, devuelve un entero que representa el punto de...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `ord` \(builtins\)](#).

2029 `OrderedDict`, clase (ucollections)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `OrderedDict`, clase \(ucollections\)](#).

2030 `clear` (ucollections.OrderedDict) — Elimina todos los elementos del objeto

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `clear` \(ucollections.OrderedDict\)](#).

2031 `copy` (ucollections.OrderedDict) — Devuelve una copia superficial del objeto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `copy` \(ucollections.OrderedDict\)](#).

2032 Método de clase `fromkeys` de Python

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Método de clase `fromkeys` de Python](#).

2033 `get` (ucollections.OrderedDict) — Devuelve el valor de la clave si la clave está en el diccionario, de lo contrario devuelve el valor...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `get` \(ucollections.OrderedDict\)](#).

2034 `items` (ucollections.OrderedDict) — Devuelve una nueva vista de los elementos del diccionario (pares (clave, valor)).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `items` \(ucollections.OrderedDict\)](#).

2035 `keys` (ucollections.OrderedDict) — Devuelve una nueva vista de las claves del diccionario

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `keys` \(ucollections.OrderedDict\)](#).

2036 `pop` (ucollections.OrderedDict) — Si la clave está en el diccionario, se elimina y se devuelve su valor; de lo contrario, se devuelve el...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `pop` \(ucollections.OrderedDict\)](#).

2037 `popitem` (ucollections.OrderedDict) — Elimina y devuelve un par (clave, valor) del diccionario.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `popitem` \(ucollections.OrderedDict\)](#).

2038 `setdefault` (ucollections.OrderedDict) — Si la clave está en el diccionario, devuelve su valor.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `setdefault` \(ucollections.OrderedDict\)](#).

2039 update (collections.OrderedDict) — Actualiza el diccionario con los pares clave/valor de otro, sobrescribiendo las claves existentes.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de update \(collections.OrderedDict\)](#).

2040 values (collections.OrderedDict) — Devuelve una nueva vista de los valores del diccionario.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de values \(collections.OrderedDict\)](#).

0873 Funciones en Python: 'P'

ID 873 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'P'

2043 pack (struct) — Empaqueta los valores de acuerdo con el parámetro `format_string` y devuelve un objeto de bytes que...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pack \(struct\)](#).

2044 pack_into (struct) — Empaqueta los valores de acuerdo con el parámetro `format_string` en un búfer que comienza en el...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pack_into \(struct\)](#).

1776 partition (builtins.bytes) — Divide la secuencia en la primera aparición de `sep`, y devuelve una tupla de 3 elementos que contiene...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de partition \(builtins.bytes\)](#).

1875 partition (builtins.str) — Divide la cadena en la primera aparición del separador `sep` y devuelve una tupla de 3 elementos que...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de partition \(builtins.str\)](#).

2184 path (sys) — Una lista mutable de directorios para buscar módulos importados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de path \(sys\)](#).

2309 Python 'pass' palabra clave

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Python 'pass' palabra clave](#).

2096 pcoeff (linalg) — Dado un vector que contiene las raíces de un polinomio, devuelve un vector que contiene los coeficientes...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pcoeff \(linalg\)](#).

2240 peektime (utimeq) — Devuelve el tiempo del elemento superior actual (es decir, el que se eliminará a continuación) pero sin...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de peektime \(utimeq\)](#).

2088 peval (linalg) — Dado un vector `p` que contiene los coeficientes de un polinomio (en orden descendente) y un valor `x`,...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de peval \(linalg\)](#).

2174 phase (cmath) — Devuelve el ángulo del número complejo `z` en radianes, conocido como la fase.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de phase \(cmath\)](#).

2082 pi (linalg) — Devuelve el valor de π utilizado por el módulo `linalg` (3.14159265358979).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pi \(linalg\)](#).

2132 pi (math) — Devuelve el valor de la constante π utilizada por el módulo matemático (3.14159265358979).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pi \(math\)](#).

2173 pi (cmath) — Devuelve el valor de la constante π utilizada por el módulo `cmath` (3.14159265358979).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pi \(cmath\)](#).

2254 pixon (hprime) — Establece el color de un solo píxel en (`x`, `y`) en el gráfico especificado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pixon \(hprime\)](#).

2270 pixon_c (hprime) — Establece el color de un solo píxel en las coordenadas cartesianas (`x`, `y`) en el gráfico especificado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pixon_c \(hprime\)](#).

2189 platform (sys) — La plataforma en la que se ejecuta MicroPython.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de platform \(sys\)](#).

2119 plot (matplotlib) — Acepta listas de coordenadas `X` e `Y`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de plot \(matplotlib\)](#).

2175 polar (cmath) — Devuelve, como una tupla, la forma polar de `z`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de polar \(cmath\)](#).

1797 pop (builtins.dict) — Si la clave está en el diccionario, se elimina y se devuelve su valor; de lo contrario, se devuelve el...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pop \(builtins.dict\)](#).

1825 pop (builtins.list) — Elimina y devuelve el elemento en el índice especificado `i` de la lista.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pop \(builtins.list\)](#).

1850 pop (builtins.set) — Elimina y devuelve un elemento arbitrario del conjunto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pop \(builtins.set\)](#).

2036 pop (ucollections.OrderedDict) — Si la clave está en el diccionario, se elimina y se devuelve su valor; de lo contrario, se devuelve el...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pop \(ucollections.OrderedDict\)](#).

2239 pop (utimeq) — Toma la entrada con el tiempo más bajo de la cola de heap.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pop \(utimeq\)](#).

1798 popitem (builtins.dict) — Elimina y devuelve un par (clave, valor) del diccionario.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de popitem \(builtins.dict\)](#).

2037 popitem (ucollections.OrderedDict) — Elimina y devuelve un par (clave, valor) del diccionario.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de popitem \(ucollections.OrderedDict\)](#).

2027 popleft (ucollections) — Elimina y devuelve un elemento del lado izquierdo del deque.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de popleft \(ucollections\)](#).

1926 pow (builtins) — Esta función calcula la potencia de un número elevando `base` a la potencia de `exp`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pow \(builtins\)](#).

2156 pow (math) — Devuelve `x` elevado a la potencia `n`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pow \(math\)](#).

2072 prevprime (arit) — Devuelve el número primo anterior menor que un entero `n`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de prevprime \(arit\)](#).

1927 print (builtins) — Esta función imprime objetos en el flujo de texto especificado (por defecto `sys.stdout`), con cada objeto...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de print \(builtins\)](#).

2198 print_exception (sys) — Imprime la excepción con un seguimiento en un objeto tipo archivo ``file`` (o ``sys.stdout`` por defecto).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de print_exception \(sys\)](#).

2100 proot (linalg) — Dado un vector que contiene los coeficientes de un polinomio, devuelve todas las raíces calculadas...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de proot \(linalg\)](#).

1832 property, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de property, clase \(builtins\)](#).

1833 getter (builtins.property) — El método getter asigna o modifica la función getter para una propiedad.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de getter \(builtins.property\)](#).

1834 setter (builtins.property) — El método setter asigna o modifica la función setter para una propiedad.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de setter \(builtins.property\)](#).

1835 deleter (builtins.property) — El método deleter asigna o modifica la función de eliminación para una propiedad.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de deleter \(builtins.property\)](#).

2238 push (utimeq) — Agrega una entrada a la cola de prioridad basada en montículo utimeq.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de push \(utimeq\)](#).

1974 pystack_use (micropython) — Devuelve el uso actual de la pila de Python (pystack) en bytes.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pystack_use \(micropython\)](#).

0874 Funciones en Python: 'Q'

ID 874 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'Q'

1972 qstr_info (micropython) — Imprime información sobre las cadenas actualmente internadas.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de qstr_info \(micropython\)](#).

0875 Funciones en Python: 'R'

ID 875 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'R'

2165 radians (math) — Devuelve los grados *x* convertidos a radianes.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de radians \(math\)](#).

2296 raise (instrucción)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de raise \(instrucción\)](#).

2104 rand (linalg) — Devuelve un vector de tamaño *n* (si *m*=0) o una matriz de forma (*n*, *m*) (si *m*>0) que contiene elementos...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rand \(linalg\)](#).

2249 randint (urandom) — Devuelve un entero aleatorio *N* tal que $a \leq N \leq b$.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de randint \(urandom\)](#).

2251 random (urandom) — Devuelve el siguiente número de punto flotante aleatorio en el rango de 0.0 a 1.0, excluyendo 1.0.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de random \(urandom\)](#).

2248 randrange (urandom) — Devuelve un número entero aleatorio de un rango.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de randrange \(urandom\)](#).

1836 range, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de range, clase \(builtins\)](#).

1986 read (uio.FileIO) — El método `read` se utiliza para leer un número especificado de bytes (``size``) de un archivo abierto (`o...`).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de read \(uio.FileIO\)](#).

1997 read (uio.TextIOWrapper) — El método `read` se utiliza para leer un número especificado de caracteres de un archivo de texto abierto...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de read \(uio.TextIOWrapper\)](#).

2007 read (uio.StringIO) — El método `read` de la clase `StringIO` se utiliza para leer datos del búfer de cadena en memoria.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de read \(uio.StringIO\)](#).

2016 read (uio.BytesIO) — El método `read` de la clase `BytesIO` se utiliza para leer datos del búfer de bytes en memoria.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de read \(uio.BytesIO\)](#).

1987 readinto (uio.FileIO) — Lee bytes en un objeto de tipo bytes preasignado y escribible, especificado por el búfer (``buffer``...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readinto \(uio.FileIO\)](#).

1998 readinto (uio.TextIOWrapper) — Lee bytes en un objeto de tipo bytes preasignado y escribible, especificado por el búfer (``buffer``...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readinto \(uio.TextIOWrapper\)](#).

2008 readinto (uio.StringIO) — Lee bytes del búfer de cadena de origen en memoria en un objeto de tipo bytes preasignado y escribible,...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readinto \(uio.StringIO\)](#).

2017 readinto (uio.BytesIO) — Lee bytes en un objeto de bytes preasignado y escribible ``buffer``, y devuelve el número de bytes leídos.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readinto \(uio.BytesIO\)](#).

1988 readline (uio.FileIO) — El método `readline` lee una línea del archivo, comenzando desde la posición actual del puntero del...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readline \(uio.FileIO\)](#).

1999 readline (uio.TextIOWrapper) — El método `readline` lee una línea del archivo de texto, comenzando desde la posición actual del puntero...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readline \(uio.TextIOWrapper\)](#).

2009 readline (uio.StringIO) — El método `readline` lee una línea del contenido del objeto `StringIO`, comenzando desde la posición actual...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readline \(uio.StringIO\)](#).

2018 readline (uio.BytesIO) — Lee una línea del contenido del objeto `BytesIO`, comenzando desde la posición actual del puntero y...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readline \(uio.BytesIO\)](#).

1989 readlines (uio.FileIO) — El método `readlines` lee múltiples líneas del archivo, comenzando desde la posición actual del puntero...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readlines \(uio.FileIO\)](#).

2000 readlines (uio.TextIOWrapper) — El método `readlines` lee múltiples líneas del archivo abierto en modo texto, comenzando desde la posición...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readlines \(uio.TextIOWrapper\)](#).

2110 real (linalg) — Devuelve un vector o matriz que contiene la parte real de x , elemento por elemento.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de real \(linalg\)](#).

2176 rect (cmath) — Devuelve el número complejo correspondiente a las coordenadas polares con módulo r y fase ϕ .

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rect \(cmath\)](#).

2267 rect (hprime) — Dibuja el contorno de un rectángulo en el gráfico especificado en (x, y) con las dimensiones dadas...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rect \(hprime\)](#).

2283 rect_c (hprime) — Dibuja el contorno de un rectángulo en el gráfico especificado en las coordenadas cartesianas $(x, ...)$

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rect_c \(hprime\)](#).

2052 red (graphic) — La constante red tiene un valor de 16711680, correspondiente al color rojo en el modelo RGB888 de 24...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de red \(graphic\)](#).

1826 remove (builtins.list) — Elimina la primera ocurrencia del elemento x de la lista.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de remove \(builtins.list\)](#).

1851 remove (builtins.set) — Elimina el elemento especificado e del conjunto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de remove \(builtins.set\)](#).

1774 replace (builtins.bytes) — Devuelve una copia de la secuencia con todas las ocurrencias de la subsecuencia old reemplazadas por...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de replace \(builtins.bytes\)](#).

1873 replace (builtins.str) — Devuelve una copia de la cadena con todas las ocurrencias de la subcadena old reemplazadas por new .

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de replace \(builtins.str\)](#).

1928 repr (builtins) — Devuelve una cadena que contiene una representación imprimible de un objeto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de repr \(builtins\)](#).

2294 return (instrucción)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de return \(instrucción\)](#).

1827 reverse (builtins.list) — Invierte los elementos de la lista en su lugar, lo que significa que la lista se modifica directamente y...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de reverse \(builtins.list\)](#).

1837 reversed (builtins) — La función `reversed` devuelve un iterador que le permite acceder a los elementos de una secuencia en...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de reversed \(builtins\)](#).

1761 rfind (builtins.bytes) — Devuelve el índice más alto en la secuencia donde se encuentra la subsecuencia `sub`, de modo que `sub` esté...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rfind \(builtins.bytes\)](#).

1860 rfind (builtins.str) — Devuelve el índice más alto en la cadena donde se encuentra la subcadena ``sub``, de modo que ``sub`` esté...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rfind \(builtins.str\)](#).

1763 rindex (builtins.bytes) — Similar a `rfind()` pero lanza un `ValueError` cuando no se encuentra la subsecuencia `sub`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rindex \(builtins.bytes\)](#).

1862 rindex (builtins.str) — Funciona como `rfind()`, pero genera un `ValueError` cuando no se encuentra la subcadena ``sub``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rindex \(builtins.str\)](#).

1929 round (builtins) — Redondea ``número`` a ``ndigits`` lugares de precisión después del punto decimal.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de round \(builtins\)](#).

1777 rpartition (builtins.bytes) — Divide la secuencia en la última aparición de ``sep`` y devuelve una tupla de 3 elementos que contiene la...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rpartition \(builtins.bytes\)](#).

1876 rpartition (builtins.str) — Divide la cadena en la última ocurrencia del separador ``sep`` y devuelve una tupla de 3 elementos que...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rpartition \(builtins.str\)](#).

2092 rref (linalg) — Devuelve la forma escalonada reducida de una matriz rectangular.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rref \(linalg\)](#).

1767 rsplit (builtins.bytes) — Divide la secuencia binaria en subsecuencias del mismo tipo, usando sep como la cadena delimitadora.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rsplit \(builtins.bytes\)](#).

1866 rsplit (builtins.str) — Divide la cadena en una lista de subcadenas, utilizando `sep` como la cadena delimitadora, comenzando...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rsplit \(builtins.str\)](#).

1772rstrip (builtins.bytes) — Devuelve una copia de la secuencia con los bytes finales especificados eliminados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rstrip \(builtins.bytes\)](#).

1871rstrip (builtins.str) — Devuelve una copia de la cadena con los caracteres finales especificados eliminados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rstrip \(builtins.str\)](#).

0876 Funciones en Python: 'S'

ID 876 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'S'

2126 scatterplot (matplotlib) — Acepta listas de coordenadas X e Y.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de scatterplot \(matplotlib\)](#).

2234 search (ure) — Esta función es parte del módulo ure, que actualmente no es compatible con el HP Prime.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de search \(ure\)](#).

2247 seed (urandom) — Inicializa el estado interno del generador de números aleatorios.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de seed \(urandom\)](#).

1991 seek (uio.FileIO) — El método seek ajusta el puntero del archivo a una ubicación específica dentro del archivo, dependiendo...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de seek \(uio.FileIO\)](#).

2002 seek (uio.TextIOWrapper) — El método seek ajusta el puntero del archivo a una ubicación específica dentro del archivo, dependiendo...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de seek \(uio.TextIOWrapper\)](#).

2011 seek (uio.StringIO) — El método seek ajusta el puntero a una ubicación específica dentro del búfer StringIO, dependiendo de...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de seek \(uio.StringIO\)](#).

2020 seek (uio.BytesIO) — El método seek ajusta el puntero a una ubicación específica dentro del búfer BytesIO, dependiendo de los...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de seek \(uio.BytesIO\)](#).

1838 set, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de set, clase \(builtins\)](#).

1839 add (builtins.set) — Agrega el elemento especificado `e` al conjunto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de add \(builtins.set\)](#).

1840 clear (builtins.set) — Elimina todos los elementos del conjunto, dejándolo vacío.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de clear \(builtins.set\)](#).

1841 copy (builtins.set) — Devuelve una copia superficial del conjunto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de copy \(builtins.set\)](#).

1843 difference (builtins.set) — Devuelve un nuevo conjunto que contiene elementos que están presentes en el conjunto original pero no en...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de difference \(builtins.set\)](#).

1844 difference_update (builtins.set) — Elimina los elementos del conjunto original que también están presentes en cualquiera de los otros...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de difference_update \(builtins.set\)](#).

1842 discard (builtins.set) — Elimina el elemento especificado `e` del conjunto si está presente.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de discard \(builtins.set\)](#).

1845 intersection (builtins.set) — Devuelve un nuevo conjunto que contiene los elementos que son comunes tanto al conjunto original como al...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de intersection \(builtins.set\)](#).

1846 intersection_update (builtins.set) — Actualiza el conjunto original para mantener solo los elementos que son comunes tanto al conjunto...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de intersection_update \(builtins.set\)](#).

1847 isdisjoint (builtins.set) — Devuelve True si el conjunto original no tiene elementos en común con el conjunto `other`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isdisjoint \(builtins.set\)](#).

1848 issubset (builtins.set) — Devuelve True si todos los elementos del conjunto original están contenidos en el conjunto `other`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de issubset \(builtins.set\)](#).

1849 issuperset (builtins.set) — Devuelve True si el conjunto original contiene todos los elementos del conjunto ``other``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de issuperset \(builtins.set\)](#).

1850 pop (builtins.set) — Elimina y devuelve un elemento arbitrario del conjunto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pop \(builtins.set\)](#).

1851 remove (builtins.set) — Elimina el elemento especificado ``e`` del conjunto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de remove \(builtins.set\)](#).

1852 symmetric_difference (builtins.set) — Devuelve un nuevo conjunto que contiene elementos que están en el conjunto original o en el conjunto...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de symmetric_difference \(builtins.set\)](#).

1853 symmetric_difference_update (builtins.set) — Actualiza el conjunto original para contener la diferencia simétrica entre el conjunto original y el...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de symmetric_difference_update \(builtins.set\)](#).

1854 union (builtins.set) — Devuelve un nuevo conjunto que contiene todos los elementos únicos del conjunto original y del conjunto...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de union \(builtins.set\)](#).

1855 update (builtins.set) — Actualiza el conjunto original añadiendo todos los elementos únicos del conjunto ``other``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de update \(builtins.set\)](#).

1908 setattr (builtins) — Este es el equivalente de `getattr()`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de setattr \(builtins\)](#).

1799 setdefault (builtins.dict) — Si la clave está en el diccionario, devuelve su valor.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de setdefault \(builtins.dict\)](#).

2038 setdefault (collections.OrderedDict) — Si la clave está en el diccionario, devuelve su valor.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de setdefault \(collections.OrderedDict\)](#).

1834 setter (builtins.property) — El método `setter` asigna o modifica la función `setter` para una propiedad.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de setter \(builtins.property\)](#).

2291 set_cartesian (hpprime) — Establece los rangos de los ejes x e y para coordenadas cartesianas.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de set_cartesian \(hpprime\)](#).

2056 draw_pixel (graphic) — Establece el color del píxel en las coordenadas (x, y) en la pantalla.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de draw_pixel \(graphic\)](#).

2242 sha256, clase (uhashlib)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de sha256, clase \(uhashlib\)](#).

2243 update (uhashlib.sha256) — Alimenta más datos binarios en el hash.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de update \(uhashlib.sha256\)](#).

2244 digest (uhashlib.sha256) — Devuelve el hash de todos los datos pasados a través del hash, como un objeto de bytes.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de digest \(uhashlib.sha256\)](#).

2105 shape (linalg) — Devuelve las dimensiones de un vector o matriz como una tupla (filas, columnas).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de shape \(linalg\)](#).

2115 show (matplotlib) — Abre un gráfico que muestra todos los objetos geométricos colocados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de show \(matplotlib\)](#).

2067 show_screen (graphic) — Congela la pantalla y espera a que se presione cualquier tecla antes de continuar.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de show_screen \(graphic\)](#).

2146 sin (math) — Devuelve el seno de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de sin \(math\)](#).

2181 sin (cmath) — Devuelve el seno de un número complejo `z`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de sin \(cmath\)](#).

2140 sinh (math) — Devuelve el seno hiperbólico de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de sinh \(math\)](#).

2106 size (linalg) — Devuelve el número de elementos en un vector o matriz como un entero.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de size \(linalg\)](#).

1856 slice, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de slice, clase \(builtins\)](#).

2091 solve (linalg) — Devuelve un vector x que satisface el sistema de ecuaciones lineales $Ax=b$.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de solve \(linalg\)](#).

1828 sort (builtins.list) — Ordena la lista en orden ascendente por defecto.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de sort \(builtins.list\)](#).

1930 sorted (builtins) — Devuelve una nueva lista que contiene todos los elementos del `iterable`, ordenados en orden ascendente...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de sorted \(builtins\)](#).

1765 split (builtins.bytes) — Divide la secuencia binaria en subsecuencias del mismo tipo, usando `sep` como la cadena delimitadora.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de split \(builtins.bytes\)](#).

1864 split (builtins.str) — Divide la cadena en una lista de subcadenas, utilizando `sep` como la cadena delimitadora.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de split \(builtins.str\)](#).

1766 splitlines (builtins.bytes) — Devuelve una lista de las líneas en la secuencia binaria, dividiéndolas en los límites de línea ASCII.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de splitlines \(builtins.bytes\)](#).

1865 splitlines (builtins.str) — Devuelve una lista de las líneas en la cadena, rompiendo en los límites de línea.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de splitlines \(builtins.str\)](#).

2133 sqrt (math) — Devuelve la raíz cuadrada de x .

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de sqrt \(math\)](#).

2180 sqrt (cmath) — Devuelve la raíz cuadrada de un número complejo `z`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de sqrt \(cmath\)](#).

1973 stack_use (micropython) — Devuelve un entero que representa el uso actual de la pila nativa en bytes.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de stack_use \(micropython\)](#).

1768 startswith (builtins.bytes) — Devuelve True si los datos binarios comienzan con el prefijo especificado, de lo contrario devuelve...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de startswith \(builtins.bytes\)](#).

1867 startswith (builtins.str) — Devuelve True si la cadena comienza con el prefijo especificado ``prefix``, de lo contrario devuelve False.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de startswith \(builtins.str\)](#).

1892 staticmethod (builtins) — Transforma un método en un método estático.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de staticmethod \(builtins\)](#).

2195 stderr (sys) — Flujo de error estándar (terminal)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de stderr \(sys\)](#).

2193 stdin (sys) — Flujo de entrada estándar (terminal)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de stdin \(sys\)](#).

2194 stdout (sys) — Flujo de salida estándar (terminal)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de stdout \(sys\)](#).

1857 str, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de str, clase \(builtins\)](#).

1858 encode (builtins.str) — Devuelve una versión codificada de la cadena como un objeto de bytes.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de encode \(builtins.str\)](#).

1859 find (builtins.str) — Devuelve el índice más bajo en la cadena donde se encuentra la subcadena ``sub``, de modo que ``sub`` esté...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de find \(builtins.str\)](#).

1860 rfind (builtins.str) — Devuelve el índice más alto en la cadena donde se encuentra la subcadena ``sub``, de modo que ``sub`` esté...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rfind \(builtins.str\)](#).

1861 index (builtins.str) — Funciona como `find()`, pero genera un `ValueError` cuando no se encuentra la subcadena ``sub``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de index \(builtins.str\)](#).

1862 rindex (builtins.str) — Funciona como `rfind()`, pero genera un `ValueError` cuando no se encuentra la subcadena ``sub``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `rindex \(builtins.str\)`](#).

1863 join (builtins.str) — Devuelve una cadena que es la concatenación de las cadenas en el iterable ``iter``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `join \(builtins.str\)`](#).

1864 split (builtins.str) — Divide la cadena en una lista de subcadenas, utilizando ``sep`` como la cadena delimitadora.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `split \(builtins.str\)`](#).

1865 splitlines (builtins.str) — Devuelve una lista de las líneas en la cadena, rompiendo en los límites de línea.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `splitlines \(builtins.str\)`](#).

1866 rsplit (builtins.str) — Divide la cadena en una lista de subcadenas, utilizando ``sep`` como la cadena delimitadora, comenzando...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `rsplit \(builtins.str\)`](#).

1867 startswith (builtins.str) — Devuelve `True` si la cadena comienza con el prefijo especificado ``prefix``, de lo contrario devuelve `False`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `startswith \(builtins.str\)`](#).

1868 endswith (builtins.str) — Devuelve `True` si la cadena termina con el sufijo especificado ``suffix``, de lo contrario devuelve `False`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `endswith \(builtins.str\)`](#).

1869 strip (builtins.str) — Devuelve una copia de la cadena con los caracteres iniciales y finales especificados eliminados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `strip \(builtins.str\)`](#).

1870 lstrip (builtins.str) — Devuelve una copia de la cadena con los caracteres iniciales especificados eliminados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `lstrip \(builtins.str\)`](#).

1871rstrip (builtins.str) — Devuelve una copia de la cadena con los caracteres finales especificados eliminados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `rstrip \(builtins.str\)`](#).

1872 format (builtins.str) — Realiza una operación de formateo de cadenas.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `format \(builtins.str\)`](#).

1873 replace (builtins.str) — Devuelve una copia de la cadena con todas las ocurrencias de la subcadena `old` reemplazadas por `new`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de `replace \(builtins.str\)`](#).

1877 center (builtins.str) — Devuelve una copia de la cadena centrada en una nueva cadena de longitud ``width``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de center \(builtins.str\)](#).

1874 count (builtins.str) — Devuelve el número de ocurrencias no superpuestas de la subcadena ``sub`` en el rango `[`start`, `end`]`.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de count \(builtins.str\)](#).

1875 partition (builtins.str) — Divide la cadena en la primera aparición del separador ``sep`` y devuelve una tupla de 3 elementos que...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de partition \(builtins.str\)](#).

1876 rpartition (builtins.str) — Divide la cadena en la última ocurrencia del separador ``sep`` y devuelve una tupla de 3 elementos que...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de rpartition \(builtins.str\)](#).

1878 lower (builtins.str) — Devuelve una copia de la cadena con todos los caracteres ASCII en mayúsculas convertidos a sus...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de lower \(builtins.str\)](#).

1879 upper (builtins.str) — Devuelve una copia de la cadena con todos los caracteres ASCII en minúsculas convertidos a su...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de upper \(builtins.str\)](#).

1880 isspace (builtins.str) — Devuelve True si todos los caracteres de la cadena son espacios en blanco ASCII y la cadena no está...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isspace \(builtins.str\)](#).

1881 isalpha (builtins.str) — Devuelve True si todos los caracteres de la cadena son caracteres ASCII alfabéticos y la cadena no está...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isalpha \(builtins.str\)](#).

1882 isdigit (builtins.str) — Devuelve True si todos los caracteres de la cadena son dígitos decimales ASCII y la cadena no está...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isdigit \(builtins.str\)](#).

1883 isupper (builtins.str) — Devuelve True si hay al menos un carácter alfabético ASCII en mayúsculas en la cadena y no hay...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de isupper \(builtins.str\)](#).

1884 islower (builtins.str) — Devuelve True si hay al menos un carácter alfabético ASCII en minúscula en la cadena y no hay caracteres...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de islower \(builtins.str\)](#).

2260 strblit (hprime) — Dibuja el gráfico_fuente en el gráfico_objetivo en (x, y), redimensionándolo a la anchura y altura...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de strblit \(hprime\)](#).

2261 strblit2 (hprime) — Dibuja un área específica de `gráfico\_fuente` en las coordenadas (`fuente\_x`, `fuente\_y`) con...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de strblit2 \(hprime\)](#).

2277 strblit2_c (hprime) — Dibuja un área específica de `gráfico\_fuente` en (`fuente\_x`, `fuente\_y`) con dimensiones `ancho\_fuente`...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de strblit2_c \(hprime\)](#).

2276 strblit_c (hprime) — Dibuja `gráfico\_fuente` en `gráfico\_objetivo` en las coordenadas cartesianas (`x`, `y`),...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de strblit_c \(hprime\)](#).

2006 StringIO, clase (uio)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de StringIO, clase \(uio\)](#).

2007 read (uio.StringIO) — El método read de la clase StringIO se utiliza para leer datos del búfer de cadena en memoria.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de read \(uio.StringIO\)](#).

2008 readinto (uio.StringIO) — Lee bytes del búfer de cadena de origen en memoria en un objeto de tipo bytes preasignado y escribible,...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readinto \(uio.StringIO\)](#).

2009 readline (uio.StringIO) — El método readline lee una línea del contenido del objeto StringIO, comenzando desde la posición actual...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readline \(uio.StringIO\)](#).

2010 write (uio.StringIO) — El método write escribe los datos especificados por el parámetro `data` (string, bytes u objeto...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de write \(uio.StringIO\)](#).

2011 seek (uio.StringIO) — El método seek ajusta el puntero a una ubicación específica dentro del búfer StringIO, dependiendo de...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de seek \(uio.StringIO\)](#).

2012 flush (uio.StringIO) — El método flush en la clase StringIO está diseñado para ser compatible con las expectativas de una...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de flush \(uio.StringIO\)](#).

2013 close (uio.StringIO) — El método close en la clase StringIO se utiliza para indicar que el objeto ya no es necesario y para...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de close \(uio.StringIO\).](#)

2014 getvalue (uio.StringIO) — Recupera todo el contenido del búfer de un objeto StringIO.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de getvalue \(uio.StringIO\).](#)

1770 strip (builtins.bytes) — Devuelve una copia de la secuencia con los bytes iniciales y finales especificados eliminados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de strip \(builtins.bytes\).](#)

1869 strip (builtins.str) — Devuelve una copia de la cadena con los caracteres iniciales y finales especificados eliminados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de strip \(builtins.str\).](#)

2109 sub (linalg) — Devuelve un vector o matriz que contiene la diferencia de A y B, elemento por elemento.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de sub \(linalg\).](#)

1931 sum (builtins) — Calcula el total sumando los elementos del objeto iterable especificado (`iterable`) al valor inicial...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de sum \(builtins\).](#)

1885 super (builtins) — Devuelve un objeto proxy que delega las llamadas a métodos a una clase padre o hermana de tipo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de super \(builtins\).](#)

1812 symmetric_difference (builtins.frozens) — Devuelve un nuevo frozenset que contiene elementos que están en este frozenset o en otro, pero no en...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de symmetric_difference \(builtins.frozens\).](#)

1852 symmetric_difference (builtins.set) — Devuelve un nuevo conjunto que contiene elementos que están en el conjunto original o en el conjunto...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de symmetric_difference \(builtins.set\).](#)

1853 symmetric_difference_update (builtins.set) — Actualiza el conjunto original para contener la diferencia simétrica entre el conjunto original y el...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de symmetric_difference_update \(builtins.set\).](#)

0877 Funciones en Python: 'T'

ID 877 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'T'

2147 tan (math) — Devuelve la tangente de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de tan \(math\)](#).

2141 tanh (math) — Devuelve la tangente hiperbólica de x.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de tanh \(math\)](#).

1992 tell (uio.FileIO) — Devuelve la posición actual del puntero de archivo dentro del archivo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de tell \(uio.FileIO\)](#).

2003 tell (uio.TextIOWrapper) — Devuelve la posición actual del puntero de archivo dentro del archivo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de tell \(uio.TextIOWrapper\)](#).

2118 text (matplotlib) — Coloca texto en el gráfico.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de text \(matplotlib\)](#).

1995 TextIOWrapper, clase (uio)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de TextIOWrapper, clase \(uio\)](#).

1996 fileno (uio.TextIOWrapper) — Devuelve el descriptor de archivo asociado con un objeto TextIOWrapper.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de fileno \(uio.TextIOWrapper\)](#).

1997 read (uio.TextIOWrapper) — El método read se utiliza para leer un número especificado de caracteres de un archivo de texto abierto...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de read \(uio.TextIOWrapper\)](#).

1998 readinto (uio.TextIOWrapper) — Lee bytes en un objeto de tipo bytes preasignado y escribible, especificado por el búfer (``buffer`` ...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readinto \(uio.TextIOWrapper\)](#).

1999 readline (uio.TextIOWrapper) — El método readline lee una línea del archivo de texto, comenzando desde la posición actual del puntero...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readline \(uio.TextIOWrapper\)](#).

2000 readlines (uio.TextIOWrapper) — El método readlines lee múltiples líneas del archivo abierto en modo texto, comenzando desde la posición...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de readlines \(uio.TextIOWrapper\)](#).

2001 write (uio.TextIOWrapper) — El método write escribe los datos especificados por el parámetro ``data`` (objeto string, bytes o...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de write \(uio.TextIOWrapper\)](#).

2002 seek (uio.TextIOWrapper) — El método seek ajusta el puntero del archivo a una ubicación específica dentro del archivo, dependiendo...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de seek \(uio.TextIOWrapper\)](#).

2003 tell (uio.TextIOWrapper) — Devuelve la posición actual del puntero de archivo dentro del archivo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de tell \(uio.TextIOWrapper\)](#).

2004 flush (uio.TextIOWrapper) — El método flush se utiliza para limpiar cualquier búfer interno del archivo y garantizar que todos los...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de flush \(uio.TextIOWrapper\)](#).

2005 close (uio.TextIOWrapper) — Cierra el archivo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de close \(uio.TextIOWrapper\)](#).

2269 textout (hprime) — Muestra el ``texto`` especificado en el ``gráfico`` en las coordenadas (``x``, ``y``) utilizando el ``color``...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de textout \(hprime\)](#).

2285 textout_c (hprime) — Muestra el texto especificado en el gráfico en las coordenadas cartesianas (x, y) utilizando el color...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de textout_c \(hprime\)](#).

2206 threshold (gc) — Establecer o consultar el umbral de asignación adicional de GC.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de threshold \(gc\)](#).

2288 ticks (hprime) — Devuelve el valor interno del reloj en milisegundos.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ticks \(hprime\)](#).

1816 to_bytes (builtins.int) — Devuelve un array de bytes que representa un entero utilizando el orden de bytes especificado.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de to_bytes \(builtins.int\)](#).

2090 transpose (linalg) — Devuelve la transpuesta de la matriz x (sin conjugación compleja).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de transpose \(linalg\)](#).

2164 trunc (math) — Devuelve un entero, siendo x redondeado hacia 0.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de trunc \(math\)](#).

2295 try (bloque try-except-else-finally)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de try \(bloque try-except-else-finally\)](#).

1886 tuple, clase (builtins)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de tuple, clase \(builtins\)](#).

1887 count (builtins.tuple) — Devuelve el número de ocurrencias del elemento value en la tupla.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de count \(builtins.tuple\)](#).

1888 index (builtins.tuple) — Devuelve el índice de la primera ocurrencia del elemento value en la tupla.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de index \(builtins.tuple\)](#).

1889 type class (builtins) — La clase incorporada type en Python cumple dos propósitos principales: identifica la clase de un objeto...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de type class \(builtins\)](#).

0878 Funciones en Python: 'U'

ID 878 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'U'

2252 uniform (urandom) — Devuelve un número de punto flotante aleatorio N tal que $a \leq N \leq b$ para $a \leq b$ y $b \leq N \leq a$ para $b < a$.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de uniform \(urandom\)](#).

1813 union (builtins.frozens) — Devuelve un nuevo frozenset con elementos de este frozenset y de todos los demás.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de union \(builtins.frozens\)](#).

1854 union (builtins.set) — Devuelve un nuevo conjunto que contiene todos los elementos únicos del conjunto original y del conjunto...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de union \(builtins.set\)](#).

2045 unpack (ustruct) — Desempaqueta los datos según la cadena_de_formato.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de unpack \(ustruct\)](#).

2046 unpack_from (ustruct) — Desempaqueta desde ``data``, comenzando en ``offset``, según la cadena de formato especificada por...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de unpack_from \(ustruct\)](#).

1800 update (builtins.dict) — Actualiza el diccionario con los pares clave/valor de otro, sobrescribiendo las claves existentes.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de update \(builtins.dict\)](#).

1855 update (builtins.set) — Actualiza el conjunto original añadiendo todos los elementos únicos del conjunto ``other``.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de update \(builtins.set\)](#).

2039 update (ucollections.OrderedDict) — Actualiza el diccionario con los pares clave/valor de otro, sobrescribiendo las claves existentes.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de update \(ucollections.OrderedDict\)](#).

2243 update (uhashlib.sha256) — Alimenta más datos binarios en el hash.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de update \(uhashlib.sha256\)](#).

1780 upper (builtins.bytes) — Devuelve una copia de la secuencia con todos los caracteres ASCII en minúsculas convertidos a su...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de upper \(builtins.bytes\)](#).

1879 upper (builtins.str) — Devuelve una copia de la cadena con todos los caracteres ASCII en minúsculas convertidos a su...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de upper \(builtins.str\)](#).

2237 clase utimeq (utimeq)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de clase utimeq \(utimeq\)](#).

2238 push (utimeq) — Agrega una entrada a la cola de prioridad basada en montículo utimeq.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de push \(utimeq\)](#).

2239 pop (utimeq) — Toma la entrada con el tiempo más bajo de la cola de heap.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pop \(utimeq\)](#).

2240 peektime (utimeq) — Devuelve el tiempo del elemento superior actual (es decir, el que se eliminará a continuación) pero sin...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de peektime \(utimeq\)](#).

0879 Funciones en Python: 'V'

ID 879 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'V'

1801 values (builtins.dict) — Devuelve una nueva vista de los valores del diccionario.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de values \(builtins.dict\)](#).

2040 values (ucollections.OrderedDict) — Devuelve una nueva vista de los valores del diccionario.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de values \(ucollections.OrderedDict\)](#).

2122 vector (matplotlib) — Crea un objeto vector que comienza en (origen_x, origen_y) y apunta en la dirección (dir_x, dir_y).

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de vector \(matplotlib\)](#).

2186 version (sys) — La versión del lenguaje Python a la que se ajusta esta implementación, como una cadena.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de version \(sys\)](#).

2187 version_info (sys) — La versión del lenguaje Python a la que se ajusta esta implementación, como una tupla de enteros.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de version_info \(sys\)](#).

0880 Funciones en Python: 'W'

ID 880 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'W'

2303 while (bucle)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de while \(bucle\)](#).

2055 white (graphic) — La constante `white` tiene un valor de 16777215, correspondiente al color blanco en el modelo RGB888 de...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de white \(graphic\)](#).

2312 Palabra clave 'with' de Python

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Palabra clave 'with' de Python](#).

1990 write (uio.FileIO) — El método write escribe los datos especificados por el parámetro `data` (bytes, bytearray o string) en...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de write \(uio.FileIO\)](#).

2001 write (uio.TextIOWrapper) — El método write escribe los datos especificados por el parámetro `data` (objeto string, bytes o...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de write \(uio.TextIOWrapper\).](#)

2010 write (uio.StringIO) — El método write escribe los datos especificados por el parámetro `data` (string, bytes u objeto...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de write \(uio.StringIO\).](#)

2019 write (uio.BytesIO) — Escribe los datos dados `data` (bytes, bytearray u objeto de cadena) en el búfer de bytes en memoria del...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de write \(uio.BytesIO\).](#)

0881 Funciones en Python: 'Y'

ID 881 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'Y'

2050 yellow (graphic) — La constante `yellow` tiene un valor de 16776960, correspondiente al color amarillo en el modelo RGB888...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de yellow \(graphic\).](#)

2307 Palabra clave 'yield' de Python

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Palabra clave 'yield' de Python.](#)

0882 Funciones en Python: 'Z'

ID 882 · título recuperado de la página española

Lista de funciones y comandos en Python que comienzan con 'Z'

2101 zeros (linalg) — Devuelve un vector (si m=0) o una matriz (si m>0) llena de ceros.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de zeros \(linalg\).](#)

1890 zip (builtins) — Crea un iterador que agrega elementos de cada uno de los iterables.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de zip \(builtins\).](#)

1933 Operadores

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operadores.](#)

1934 Operadores lógicos

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operadores lógicos.](#)

2317 Operador lógico AND

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador lógico AND.](#)

2318 Operador lógico OR

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador lógico OR.](#)

2319 Operador lógico NOT

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador lógico NOT.](#)

1935 Operadores aritméticos

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operadores aritméticos.](#)

2320 Operador de adición (+)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador de adición \(+\).](#)

2321 Operador de resta (-)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador de resta \(-\).](#)

2322 Operador de multiplicación (*)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador de multiplicación \(*\).](#)

2323 Operador de división (/)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador de división \(/\).](#)

2324 Operador de división entera (//)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador de división entera \(//\).](#)

2325 Operador de módulo (%)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador de módulo \(%\).](#)

2326 Operador de exponenciación (**)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador de exponenciación \(**\).](#)

1936 Operadores de comparación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operadores de comparación.](#)

2327 Operador de igualdad (==)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador de igualdad \(==\).](#)

2328 Operador de desigualdad (!=)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador de desigualdad \(!=\)](#).

2329 Operador mayor que (>)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador mayor que \(>\)](#).

2330 Operador menor que (<)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador menor que \(<\)](#).

2331 Operador mayor o igual (>=)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador mayor o igual \(>=\)](#).

2332 Operador menor o igual (<=)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador menor o igual \(<=\)](#).

1937 Operadores a nivel de bits

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operadores a nivel de bits](#).

2335 Operador AND-bit (&)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador AND-bit \(&\)](#).

2336 Operador OR-bit (|)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador OR-bit \(|\)](#).

2337 Operador XOR-bit (^)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador XOR-bit \(^\)](#).

2338 Operador NOT-bit (~)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador NOT-bit \(~\)](#).

2333 Operador despl-izquierda (<<)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador despl-izquierda \(<<\)](#).

2334 Operador despl-derecha (>>)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador despl-derecha \(>>\)](#).

1938 Operadores de asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operadores de asignación](#).

2339 Operador de asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador de asignación.](#)

2340 Operador suma-asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador suma-asignación.](#)

2341 Operador resta-asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador resta-asignación.](#)

2342 Operador multiplicación-asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador multiplicación-asignación.](#)

2343 Operador división-asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador división-asignación.](#)

2344 Operador división-piso-asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador división-piso-asignación.](#)

2345 Operador módulo-asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador módulo-asignación.](#)

2346 Operador exponenciación-asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador exponenciación-asignación.](#)

2347 Operador AND-bit-asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador AND-bit-asignación.](#)

2348 Operador OR-bit-asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador OR-bit-asignación.](#)

2349 Operador XOR-bit-asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador XOR-bit-asignación.](#)

2350 Operador despl-izquierda-asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador despl-izquierda-asignación.](#)

2351 Operador despl-derecha-asignación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Operador despl-derecha-asignación.](#)

1939 %-operador de formateo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de %-operador de formateo.](#)

1940 Excepciones

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Excepciones.](#)

1941 BaseException, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de BaseException, clase.](#)

1942 ArithmeticError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ArithmeticError, clase.](#)

1943 AssertionError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de AssertionError, clase.](#)

1944 AttributeError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de AttributeError, clase.](#)

1945 EOFError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de EOFError, clase.](#)

1946 Exception, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Exception, clase.](#)

1947 GeneratorExit, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de GeneratorExit, clase.](#)

1948 ImportError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ImportError, clase.](#)

1949 IndentationError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de IndentationError, clase.](#)

1950 IndexError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de IndexError, clase.](#)

1951 KeyboardInterrupt, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de KeyboardInterrupt, clase.](#)

1952 KeyError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de KeyError, clase.](#)

1953 LookupError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de LookupError, clase.](#)

1954 MemoryError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de MemoryError, clase.](#)

1955 NameError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de NameError, clase.](#)

1956 NotImplementedError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de NotImplementedError, clase.](#)

1957 OSError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de OSError, clase.](#)

1958 OverflowError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de OverflowError, clase.](#)

1959 RuntimeError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de RuntimeError, clase.](#)

1960 StopAsyncIteration, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de StopAsyncIteration, clase.](#)

1961 StopIteration, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de StopIteration, clase.](#)

1962 SyntaxError, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de SyntaxError, clase.](#)

1963 **SystemExit**, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de SystemExit, clase.](#)

1964 **TypeError**, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de TypeError, clase.](#)

1965 **UnicodeError**, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de UnicodeError, clase.](#)

1966 **ValueError**, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ValueError, clase.](#)

1967 **ZeroDivisionError**, clase

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ZeroDivisionError, clase.](#)

0526 Menús del cuadro de herramientas

ID 526 · título localizado en HelpEs

Pulse la tecla del cuadro de herramientas para acceder a los menús que contienen los comandos y funciones más comunes de la calculadora.

- [[Matem.]] enumera las funciones matemáticas más comunes
- [[Sistema]] enumera las funciones simbólicas más comunes
- [[Apl.]] enumera las funciones específicas de aplicaciones
- [[Usua.]] enumera las funciones y programas que ha creado usted mismo
- [[Catlg]] enumera todas las funciones y comandos

Puede seleccionar una opción pulsándola o utilizando la rueda para resaltarla y, a continuación, presionar Enter. En un menú de opciones, también puede seleccionar una entrada por su número o escribiendo la primera letra, o dos letras, de su nombre y luego presionar la tecla Enter.

También puede utilizar una función o comando letra a letra.

Modo de presentación de menús:

El modo de presentación de menús predeterminado muestra nombres descriptivos para las funciones Matem. y Sistema. Por ejemplo, "ifactors" se representa como "Lista de factores". Si prefiere que las funciones se presenten por su nombre de comando, desactive la opción Pantalla del menú en la segunda página de la pantalla Configuración de Inicio.

0527 Menú Matem.

ID 527 · título localizado en HelpEs

El menú Matemáticas del cuadro de herramientas enumera las funciones matemáticas más comunes.

0528 Números

ID 528 · título localizado en HelpEs

Este menú enumera las funciones básicas de números reales

0529 Techo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Techo.](#)

0530 Suelo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Suelo.](#)

0531 Parte entera

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Parte entera.](#)

0532 Parte fraccional

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Parte fraccional.](#)

0533 Redondear

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Redondear.](#)

0534 Truncar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Truncar.](#)

0535 Mantis

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Mantis.](#)

0536 Exponente

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Exponente.](#)

0537 Aritmética

ID 537 · título localizado en HelpEs

Este menú enumera las funciones básicas de aritmética.

0538 Máximo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Máximo.](#)

0539 Mínimo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Mínimo.](#)

0540 Módulo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Módulo.](#)

0541 Encontrar raíz

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Encontrar raíz.](#)

0542 Porcentaje

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Porcentaje.](#)

0543 Compleja

ID 543 · título localizado en HelpEs

Este menú incluye las funciones numéricas complejas básicas

0544 Argumento

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Argumento.](#)

0545 Complejo conjugado

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Complejo conjugado.](#)

0546 Parte imaginaria

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Parte imaginaria.](#)

0547 Parte real

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Parte real.](#)

0548 Vector de la unidad o signo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Vector de la unidad o signo.](#)

0549 Exp y Ln

ID 549 · título localizado en HelpEs

Este menú enumera las funciones de registro y exponenciales básicas

0550 Antilogaritmo común

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Antilogaritmo común.](#)

0551 Exponente menos 1

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Exponente menos 1.](#)

0552 Logaritmo natural más 1

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Logaritmo natural más 1.](#)

0553 Trigonometría

ID 553 · título localizado en HelpEs

Este menú enumera las funciones básicas de trigonometría

0554 Cosecante

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Cosecante.](#)

0555 Arco cosecante

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Arco cosecante.](#)

0556 Secante

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Secante.](#)

0557 Arco secante

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Arco secante.](#)

0558 Cotangente

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Cotangente.](#)

0559 Arco cotangente

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Arco cotangente.](#)

0560 Hiperbólica

ID 560 · título localizado en HelpEs

Este menú enumera las funciones hiperbólicas básicas

0561 Seno hiperbólico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Seno hiperbólico.](#)

0562 Seno hiperbólico inverso

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Seno hiperbólico inverso.](#)

0563 Coseno hiperbólico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Coseno hiperbólico.](#)

0564 Coseno hiperbólico inverso

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Coseno hiperbólico inverso.](#)

0565 Tangente hiperbólica

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tangente hiperbólica.](#)

0566 Tangente hiperbólica inversa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tangente hiperbólica inversa.](#)

0567 Probabilidad

ID 567 · título localizado en HelpEs

Este menú incluye las funciones básicas de probabilidad

0568 Factorial

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Factorial.](#)

0569 Combinaciones

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Combinaciones.](#)

0570 Permutaciones

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Permutaciones.](#)

1742 Asistente de Probabilidad

ID 1742 · título localizado en HelpEs

El asistente de probabilidad te proporciona una forma gráfica de usar las funciones de distribución de probabilidad.

Selecciona el tipo de probabilidad que quieres utilizar en la parte superior.

Lo que quieres que tus cálculos se centren (centro, colas, superior/inferior...)

Introduce los datos y parámetros conocidos.

Luego puedes seleccionar ya sea la probabilidad o los límites y presionar **[[Soluc.]]** para realizar el cálculo.

0571 Aleatorio

ID 571 · título localizado en HelpEs

Este menú contiene funciones neméricas aleatorias.

0572 Número aleatorio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Número aleatorio.](#)

0573 Entero aleatorio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Entero aleatorio.](#)

0574 Normal aleatorio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Normal aleatorio.](#)

0576 Valor de inicialización aleatorio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Valor de inicialización aleatorio.](#)

0577 Densidad

ID 577 · título localizado en HelpEs

Este menú incluye las funciones de distribución de densidad

0578 Densidad normal

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Densidad normal.](#)

0579 Densidad T de Student

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Densidad T de Student.](#)

0580 Densidad χ^2

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Densidad \$\chi^2\$.](#)

0581 Densidad de Fisher

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Densidad de Fisher.](#)

0582 Densidad de probabilidad binomial

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Densidad de probabilidad binomial.](#)

0583 Densidad geométrica

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Densidad geométrica.](#)

0584 Densidad de Poisson

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Densidad de Poisson.](#)

0585 Acumulado

ID 585 · título localizado en HelpEs

Este menú incluye las funciones de distribución acumulada.

0586 Normal acumulada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Normal acumulada.](#)

0587 T de Student acumulada

ID 587 · título recuperado de la página española

Sintaxis

STUDENT_CDF(d, x [, x2])

Función de distribución t de Student acumulada

Con dos valores (d y x), devuelve la probabilidad con cola más baja de la función de densidad de probabilidad del Estudiante a x, dados d grados de libertad. Con el tercer argumento opcional x2, devuelve el área debajo de la función de densidad de probabilidad del Estudiante entre los dos valores de x.

0588 Acumulada χ^2

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Acumulada \$\chi^2\$.](#)

0589 Fisher acumulada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Fisher acumulada.](#)

0590 Binomial acumulada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Binomial acumulada.](#)

0591 Geométrica acumulada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Geométrica acumulada.](#)

0592 Poisson acumulada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Poisson acumulada.](#)

0593 Inverso

ID 593 · título localizado en HelpEs

Este menú incluye las funciones de distribución acumulada inversa

0594 Normal acumulada inversa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Normal acumulada inversa.](#)

0595 T de Student acumulada inversa

ID 595 · título recuperado de la página española

Sintaxis

STUDENT_ICDF(d, p)

Función de distribución t de Student acumulada inversa.

Devuelve el valor de x de forma que la probabilidad de cola inferior t de Student de x, con d grados de libertad, sea p.

0596 Acumulada inversa χ^2

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Acumulada inversa \$\chi^2\$.](#)

0597 Fisher acumulada inversa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Fisher acumulada inversa.](#)

0599 Binomial acumulada inversa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Binomial acumulada inversa.](#)

0600 Geométrica acumulada inversa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Geométrica acumulada inversa.](#)

0601 Inversa de Poisson acumulada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Inversa de Poisson acumulada.](#)

0602 Lista

ID 602 · título localizado en HelpEs

Este menú enumera las funciones básicas de lista

0603 Crear lista

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Crear lista.](#)

0604 Ordenar lista

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ordenar lista.](#)

0605 Invertir lista

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Invertir lista.](#)

0606 Concatenar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Concatenar.](#)

1648 Eliminar elementos

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Eliminar elementos.](#)

1649 INSERT

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de INSERT.](#)

0607 Posición

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Posición.](#)

0608 Tamaño de la lista

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tamaño de la lista.](#)

0609 Lista Δ

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Lista \$\Delta\$.](#)

0610 Lista Σ

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Lista \$\Sigma\$.](#)

0611 Lista Π

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Lista \$\Pi\$.](#)

0612 Diferencia de lista

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Diferencia de lista.](#)

0613 Lista de Intersección

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Lista de Intersección.](#)

0614 Lista de Union

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Lista de Union.](#)

1518 EQ — Devuelve 1 si los dos objetos son lo mismo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de EQ.](#)

0615 Matriz

ID 615 · título localizado en HelpEs

Este menú enumera las funciones matriciales básicas

0616 Transposición

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Transposición.](#)

0617 Determinante de la matriz cuadrada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Determinante de la matriz cuadrada.](#)

0618 Matriz escalonada reducida

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Matriz escalonada reducida.](#)

0619 Crear

ID 619 · título localizado en HelpEs

Este menú enumera las funciones de creación de matrices

0620 Crear matriz

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Crear matriz.](#)

0621 Matriz de identidad

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Matriz de identidad.](#)

0622 Matriz aleatoria

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Matriz aleatoria.](#)

0623 Jordan Block

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Jordan Block.](#)

0624 Matriz de Hilbert

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Matriz de Hilbert.](#)

0625 Isometría

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Isometría.](#)

0626 Matriz de Vandermonde

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Matriz de Vandermonde.](#)

0627 Básico

ID 627 · título localizado en HelpEs

Este menú enumera las funciones matriciales básicas

0628 Valor de absoluto

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Valor de absoluto.](#)

0629 Norma de fila

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Norma de fila.](#)

0630 Norma de columna

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Norma de columna.](#)

0631 Norma espectral

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Norma espectral.](#)

0632 Radio espectral

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Radio espectral.](#)

0633 Número de condición

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Número de condición.](#)

0634 Clasificación de matriz rectangular

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Clasificación de matriz rectangular.](#)

0635 Pivote

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Pivote.](#)

0636 Trazar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar.](#)

0637 Opciones avanzadas

ID 637 · título localizado en HelpEs

Este menú enumera las funciones matriciales avanzadas.

0638 Valores propios

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Valores propios.](#)

0639 Valores y vectores propios

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Valores y vectores propios.](#)

0640 Jordan

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Jordan.](#)

0642 Diagonal

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Diagonal.](#)

0643 Cholesky

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Cholesky.](#)

0644 Normal de Hermite

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Normal de Hermite.](#)

0645 Hessenberg

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Hessenberg.](#)

0646 Normal de Smith

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Normal de Smith.](#)

0648 Factorizar

ID 648 · título localizado en HelpEs

Este menú enumera las funciones matriciales de factorización

0649 Factorización LQ

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Factorización LQ.](#)

0650 Mínimos cuadrados

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Mínimos cuadrados.](#)

0651 Descomposición LU

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Descomposición LU.](#)

0652 Factorización QR

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Factorización QR.](#)

0653 Descomposición Schur

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Descomposición Schur.](#)

0654 Descomposición de valor único

ID 654 · título recuperado de la página española

Sintaxis

`SVD(matriz)`

Factoriza una matriz $m \times n$ en dos matrices ortogonales, U ($m \times m$) y V ($n \times n$), y un vector S como la matriz $= U \cdot S' \cdot \text{trn}(V)$. (S' es la diagonalización de S .)

0655 Valores únicos

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Valores únicos.](#)

0656 Vector

ID 656 · título localizado en HelpEs

Este menú enumera las funciones vectoriales

0657 Producto vectorial

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Producto vectorial.](#)

0658 Producto escalar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Producto escalar.](#)

0659 Norma máx.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Norma máx..](#)

0660 Norma L^1

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Norma \$L^1\$.](#)

0661 Norma L^2

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Norma \$L^2\$.](#)

0662 Especial

ID 662 · título localizado en HelpEs

Este menú enumera las funciones especiales vinculadas a las funciones de distribución

0663 Beta — Devuelve el valor de la función Beta para dos valores, x e y , definido como $\text{Gamma}(x) \cdot \text{Gamma}(y) / \text{Gamma}(x+y)$.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Beta](#).

0664 Función de error

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función de error](#).

0665 Función de Error complementaria

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función de Error complementaria](#).

0666 Función Gamma

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función Gamma](#).

0667 Psi — Devuelve el valor de la derivada n th de la función digamma en $x=a$, donde la función digamma es la...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Psi](#).

0668 Zeta — Devuelve el valor de la función zeta (Z) para un número x real.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Zeta](#).

0669 Integral del coseno

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Integral del coseno](#).

0670 Integral exponencial

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Integral exponencial](#).

0671 Integral del seno

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Integral del seno](#).

0672 Menú CAS

ID 672 · título localizado en HelpEs

El menú CAS del cuadro de herramientas enumera las funciones más útiles del sistema algebraico computacional (CAS).

0673 Álgebra

ID 673 · título localizado en HelpEs

El menú Álgebra contiene comandos de álgebra simbólica comunes, como collect, expand y factor.

0674 Simplificar expresión

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Simplificar expresión.](#)

0675 Recoger como términos

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Recoger como términos.](#)

0676 Ampliar expresión

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ampliar expresión.](#)

0677 Factorizar polinomio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Factorizar polinomio.](#)

0678 Descomposición de fracción parcial

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Descomposición de fracción parcial.](#)

0679 Sustituir

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Sustituir.](#)

0680 Extraer

ID 680 · título localizado en HelpEs

Este menú contiene comandos que le permiten extraer un lado de una ecuación o una parte de una fracción.

0681 Denominador simplificado

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Denominador simplificado.](#)

0682 Numerador simplificado

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Numerador simplificado.](#)

0683 Lado izquierdo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Lado izquierdo.](#)

0684 Lado derecho

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Lado derecho.](#)

0685 Cálculos

ID 685 · título localizado en HelpEs

El menú Cálculos contiene operaciones relativas a los límites, diferenciación e integración.

0686 Diferenciar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Diferenciar.](#)

0688 Límite

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Límite.](#)

0687 Integrar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Integrar.](#)

0689 Ampliación de serie

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ampliación de serie.](#)

0690 Sumatorio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Sumatorio.](#)

0691 Diferencial

ID 691 · título localizado en HelpEs

Este submenú contiene operaciones vectoriales especializadas basadas en la diferenciación, como rotacional o gradiente, así como las transformaciones Laplace y Laplace inverso.

0692 Llave rotacional

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Llave rotacional.](#)

0693 Divergencia

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Divergencia.](#)

0694 Gradiente

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Gradiente.](#)

0695 Matriz hessiana

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Matriz hessiana.](#)

0696 Integral

ID 696 · título localizado en HelpEs

Este menú contiene operaciones especializadas basadas en las integración, como la integración por partes.

0697 Integración por partes v

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Integración por partes v.](#)

0698 Integración por partes u

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Integración por partes u.](#)

0699 preval — Devuelve $F(b) - F(a)$.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de preval.](#)

0700 Límites

ID 700 · título localizado en HelpEs

Este submenú contiene operaciones especializadas con límites, como los polinomios de Taylor.

0701 Ampliación de serie de Taylor

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ampliación de serie de Taylor.](#)

0702 Taylor de cociente

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Taylor de cociente.](#)

0703 Suma de Riemann

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Suma de Riemann.](#)

0704 Transformar

ID 704 · título localizado en HelpEs

Este menú contiene los comandos de transformadas de Laplace y Fourier.

0705 Transformada de Laplace

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Transformada de Laplace.](#)

0706 Transformada de Laplace inversa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Transformada de Laplace inversa.](#)

0707 Transformada rápida de Fourier

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Transformada rápida de Fourier.](#)

0708 Transformada de Fourier rápida Inversa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Transformada de Fourier rápida Inversa.](#)

0709 Soluc.

ID 709 · título localizado en HelpEs

El menú Soluc. contiene los diversos comandos para resolver ecuaciones.

0710 Solución de CAS

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Solución de CAS.](#)

0711 Ceros

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ceros.](#)

0712 Solución de complejo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Solución de complejo.](#)

0714 Ceros complejos

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ceros complejos.](#)

0715 Solución numérica

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Solución numérica.](#)

0716 Resolver ecuación diferencial

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Resolver ecuación diferencial.](#)

0717 ODE Solver

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ODE Solver.](#)

0718 Solucionador de sistema lineal

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Solucionador de sistema lineal.](#)

0719 Reescribir

ID 719 · título localizado en HelpEs

El menú Reescribir contiene comandos para reescribir o simplificar expresiones utilizando una variedad de promedios, incluidas las identidades trigonométricas, etc.

0720 Recopilar logaritmos

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Recopilar logaritmos.](#)

0721 Ampliar potencia

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ampliar potencia.](#)

0722 Ampliación trascendental

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ampliación trascendental.](#)

0723 Exp y Log

ID 723 · título localizado en HelpEs

Este menú contiene comandos para convertir expresiones en expresiones equivalentes utilizando varias identidades que impliquen potencias, logaritmos y exponentes.

0724 exp2pow — Transforma una expresión de la forma $e^{(n \ln(x))}$ reescrita como una potencia de x .

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de exp2pow.](#)

0725 pow2exp — Devuelve una expresión con potencias reescritas como un exponencial.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de pow2exp.](#)

0726 exp2trig — Devuelve una expresión con complejos exponenciales reescritos en términos de seno y coseno.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de exp2trig.](#)

0727 Ampliar exponenciales

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ampliar exponenciales.](#)

0728 Sen a ...

ID 728 · título localizado en HelpEs

Esté menú contiene comandos para convertir expresiones que contienen la función inversa de seno en expresiones equivalentes que contienen otras funciones trigonométricas inversas.

0729 asin2acos — Reemplaza $\arcsin(x)$ por $\pi/2 - \arccos(x)$ en Expr.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de asin2acos.](#)

0730 asin2atan — Sustituye $\arcsin(x)$ por $\arctan(x/\sqrt{1-x^2})$ en Expr.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de asin2atan.](#)

0731 sin2costan — Reescribe Expr de forma que $\sin(x)$ se reemplaza por $\cos(x) \cdot \tan(x)$

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de sin2costan.](#)

0732 Cos a ...

ID 732 · título localizado en HelpEs

Esté menú contiene comandos para convertir expresiones que contienen la función inversa de coseno en expresiones equivalentes que contienen otras funciones trigonométricas inversas.

0733 acos2asin — Reemplaza $\arccos(x)$ por $\pi/2 - \arcsin(x)$ en el argumento Expr.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de acos2asin.](#)

0734 acos2atan — Sustituye $\arccos(x)$ por $\pi/2 - \arctan(x/\sqrt{1-x^2})$ en el argumento.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de acos2atan.](#)

0735 cos2sintan — Reemplaza $\cos(x)$ por $\sin(x)/\tan(x)$ en el argumento.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de cos2sintan.](#)

0736 Tan a ...

ID 736 · título localizado en HelpEs

Esté menú contiene comandos para convertir expresiones que contienen la función inversa de tangente en expresiones equivalentes que contienen otras funciones trigonométricas inversas.

0737 atan2asin — Sustituye $\arctan(x)$ por $\arcsin(x/\sqrt{1+x^2})$ en el argumento Expr.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de atan2asin.](#)

0738 atan2acos — Sustituye $\arctan(x)$ por $\pi/2 - \arccos(x/\sqrt{1+x^2})$ en el argumento.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de atan2acos.](#)

0739 tan2sincos — Reescribe Expr con $\tan(x)$ usando $\sin(x)/\cos(x)$

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de tan2sincos.](#)

0740 halftan — Transforma $\sin(x)$, $\cos(x)$ y $\tan(x)$ como una función de $\tan(x/2)$.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de halftan](#).

0742 Trigonometría a ...

ID 742 · título localizado en HelpEs

Este menú contiene comandos para convertir las expresiones que contienen las diversas funciones transcendentales en expresiones equivalentes que contengan otras funciones transcendentales.

0743 trigsin — Devuelve una expresión simplificada mediante las fórmulas $\sin(x)^2 + \cos(x)^2 = 1$ y $\tan(x) = \sin(x)/\cos(x)$.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de trigsin](#).

0744 trigcos — Devuelve una expresión simplificada mediante las fórmulas $\sin(x)^2 + \cos(x)^2 = 1$ y $\tan(x) = \sin(x)/\cos(x)$.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de trigcos](#).

0745 trigtan — Devuelve una expresión simplificada mediante las fórmulas $\sin(x)^2 + \cos(x)^2 = 1$ y $\tan(x) = \sin(x)/\cos(x)$ Se da...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de trigtan](#).

0746 atrig2ln — Devuelve una expresión con las funciones de trigonométricas inversas usando la función de logaritmo...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de atrig2ln](#).

0747 tlin — Devuelve una expresión trigonométrica con los productos y las potencias de enteros linealizados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de tlin](#).

0748 tcollect — Devuelve una expresión trigonométrica linealizada y con cualquier término de seno y coseno del mismo...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de tcollect](#).

0749 trigexpand — Devuelve una expresión trigonométrica en forma expandida.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de trigexpand](#).

0750 trig2exp — Devuelve una expresión con las funciones trigonométricas reescritas como complejos exponenciales (sin...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de trig2exp](#).

0751 Entero

ID 751 · título localizado en HelpEs

El menú Enteros contiene operaciones con enteros.

0752 Divisores de enteros

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Divisores de enteros](#).

0753 Factores de entero

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Factores de entero](#).

0754 Lista de factores de enteros

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Lista de factores de enteros.](#)

0755 DCM de entero

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de DCM de entero.](#)

0756 Múltiplo común mínimo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Múltiplo común mínimo.](#)

0757 Primo

ID 757 · título localizado en HelpEs

El submenú Primo contiene operaciones relacionadas con números primos.

0758 Prueba de primalidad de enteros

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Prueba de primalidad de enteros.](#)

0759 N-ésimo primo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de N-ésimo primo.](#)

0760 Siguiente primo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Siguiente primo.](#)

0761 Primo anterior

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Primo anterior.](#)

0762 Totiente de Euler

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Totiente de Euler.](#)

0763 División

ID 763 · título localizado en HelpEs

El submenú División contiene operaciones relacionadas con la división con enteros.

0764 Cociente euclidiano de entero

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Cociente euclidiano de entero.](#)

0765 Resto euclidiano entero

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Resto euclidiano entero.](#)

0766 Módulo y potencia de entero

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Módulo y potencia de entero.](#)

0767 Resto chino con enteros

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Resto chino con enteros.](#)

0768 Polinómica

ID 768 · título localizado en HelpEs

El submenú Polinómica contiene comandos relacionados con los polinomios.

0781 Álgebra

ID 781 · título localizado en HelpEs

Entre otras operaciones, este menú contiene los equivalentes polinómicos de algunos de los comandos que se encuentran en el menú Enteros.

0782 Cociente

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Cociente.](#)

0783 Resto

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Resto.](#)

0784 Grado de un polinomio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Grado de un polinomio.](#)

0785 Factor por grado

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Factor por grado.](#)

0786 Coeficiente de GCD

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Coeficiente de GCD.](#)

0787 Recuento de ceros

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Recuento de ceros.](#)

0788 Resto chino

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Resto chino.](#)

0774 Crear

ID 774 · título localizado en HelpEs

Este submenú contiene comandos para crear polinomios, de forma aleatoria o con propiedades específicas.

0775 Polinomio a coeficientes

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Polinomio a coeficientes.](#)

0776 Coeficientes a polinomio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Coeficientes a polinomio.](#)

0777 Raíces a coeficientes

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Raíces a coeficientes.](#)

0778 Raíces a polinomio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Raíces a polinomio.](#)

0779 Polinomio aleatorio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Polinomio aleatorio.](#)

0780 Polinomio mínimo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Polinomio mínimo.](#)

0789 Especial

ID 789 · título localizado en HelpEs

Este submenú proporciona acceso a comandos especializados relacionados con funciones polinómicas.

0790 Ciclotómico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ciclotómico.](#)

0791 Bases de Groebner

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Bases de Groebner.](#)

0792 Groebner Remainder (Resto de Groebner)

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Groebner Remainder \(Resto de Groebner\).](#)

0793 Polinomio de Hermite

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Polinomio de Hermite.](#)

0794 Polinomio de Lagrange

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Polinomio de Lagrange.](#)

0795 Polinomio de Laguerre

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Polinomio de Laguerre.](#)

0796 Polinomio de Legendre

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Polinomio de Legendre.](#)

1735 ListToMat — Convierte una lista en un vector o una lista de listas en una matriz.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ListToMat.](#)

1712 Simplificación LLL

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Simplificación LLL.](#)

1692 Sin operación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Sin operación.](#)

0797 Tn de Chebyshev

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tn de Chebyshev.](#)

0798 Un de Chebyshev

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Un de Chebyshev.](#)

0770 Coeficientes del polinomio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Coeficientes del polinomio.](#)

0771 Divisores de polinomios

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Divisores de polinomios.](#)

0772 Lista de factor polinomial

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Lista de factor polinomial.](#)

0769 Raíces de polinomios

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Raíces de polinomios.](#)

0773 Divisor común máximo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Divisor común máximo.](#)

0799 Divisor común máximo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Divisor común máximo.](#)

1707 groupermu — Devuelve el grupo de permutaciones generadas por permutación1 y permutación2.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de groupermu.](#)

1699 Función is_cycle

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función is_cycle.](#)

1698 Función is_permu

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función is_permu.](#)

0756 Múltiplo común mínimo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Múltiplo común mínimo.](#)

0800 Gráfico

ID 800 · título localizado en HelpEs

El menú Gráfico contiene operaciones que permiten el dibujo de gráficos en el sistema algebraico computacional.

0801 Trazar contorno

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar contorno.](#)

0173 Trazar función

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar función.](#)

0177 Trazar implícito

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar implícito.](#)

0179 Trazar campo de direcciones

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar campo de direcciones.](#)

0180 Trazar EDO

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar EDO.](#)

0181 Trazar lista

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Trazar lista.](#)

0802 Menú Apl.

ID 802 · título localizado en HelpEs

El menú Aplicaciones del cuadro de herramientas enumera todas las funciones específicas de la aplicación.

0803 Menú Usua.

ID 803 · título localizado en HelpEs

El menú Usua. enumera todas las funciones y todos los programas que ha creado. Se agruparán con el nombre del archivo de origen que contiene las variables o funciones exportadas.

0804 Menú Catlg.

ID 804 · título localizado en HelpEs

El menú Catálogo del Cuadro de herramientas enumera todas las funciones y comandos del sistema.

Al lado derecho del encabezado del catálogo hay un icono de información pequeño (i). Pulse el icono para ver el número de cada tipo de función definida actualmente en su HP Prime (CAS, Apl., Usua., etcétera).

1345 Unidades

ID 1345 · título localizado en HelpEs

Puede agregar unidades a los números para transformarlos en medidas. También puede recuperar los valores de muchas constantes físicas comunes. Estos recursos están disponibles al presionar Shift Unids. Aparecen los siguientes botones del menú:

Botones de menú:

- `[[Herramientas]]` para realizar determinadas operaciones relativas a medidas
- `[[Unids]]` abre un menú con 18 submenús, cada uno de los cuales proporciona unidades con un dominio particular (área, volumen, electricidad, etc.)
- `[[Const]]` abre un menú con 4 submenús, cada uno de los cuales proporciona constantes con un dominio particular (química, física, mecánica cuántica, etc.)
- `[[Valor]]` seleccione si desea recuperar el valor de una constante o su nombre
- `[[OK]]` copia el comando seleccionado en la línea de entrada

Presione la tecla Esc para cerrar el menú sin realizar ninguna selección.

1346 Herramientas de unidad

ID 1346 · título localizado en HelpEs

Hay disponibles cuatro herramientas para manipular las medidas:

- **CONVERT:** permite realizar conversiones entre unidades
- **MKSA:** sistema de metros, kilogramos, segundos y amperios
- **UFACTOR:** permite factorizar unidades
- **USIMPLIFY:** permite simplificar unidades.

0953 Convertir

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Convertir.](#)

1143 Convertir al sistema métrico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Convertir al sistema métrico.](#)

1310 Conversión de factores de unidades

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Conversión de factores de unidades.](#)

1311 Simplificación de unidades

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Simplificación de unidades.](#)

1347 Unidades

ID 1347 · título localizado en HelpEs

Presione Shift Unids para abrir el Menú unidades y luego pulse en el botón del `[[Unids]]` menú. El menú cuenta con 18 submenús y cada uno ofrece las unidades desde un dominio particular (área, volumen, electricidad, etc.).

Si agrega una unidad a un número, convertirá dicho número en una medida.

Ciertas operaciones aritméticas se pueden realizar sobre medidas, incluso si la medida cuenta con unidades diferentes. (No obstante, las unidades deben ser consistentes: todas deben ser de longitud, volumen, etc.). Por ejemplo, para agregar 5 cm a 2 pulgadas:

- Introduzca 5 en la línea de entrada.
- Presione Shift Unids.
- Cuando se abra el menú [[Unids]], toque Longitud y seleccione cm. 5_cm aparece en la línea de entrada. Tenga en cuenta el carácter de guión bajo.
- Introduzca + 2.
- Presione Shift Units.
- Cuando se abra el menú [[Unids]], toque Longitud y seleccione la opción de pulgadas.
- Presione Enter. El resultado es 10.08_cm.

Tenga que en cuenta que la unidad de la primera medida introducida se utilizará para el resultado.

PREFIXES

Tenga en cuenta que el primer elemento del menú Unids es Prefijo. Los prefijos no son unidades, sino multiplicadores. Seleccione el prefijo que desee antes de seleccionar la unidad.

1348 Prefijo

ID 1348 · título localizado en HelpEs

El menú Unids incluye una entrada que no es una categoría de unidad, denominada Prefijo. La selección de esta opción mostrará una paleta de prefijos.

Y: yota 10^{24}
 Z: zeta 10^{21}
 E: exa 10^{18}
 P: peta 10^{15}
 T: tera 10^{12}
 G: giga 10^9
 M: mega 10^6
 k: kilo 10^3
 h: hecto 10^2
 D: deca 10
 d: deci 10^{-1}
 c: centi 10^{-2}
 m: mili 10^{-3}
 μ: micro 10^{-6}
 n: nano 10^{-9}
 p: pico 10^{-12}
 f: femto 10^{-15}
 a: atto 10^{-18}
 z: zepto 10^{-21}
 y: yocto 10^{-24}

Nota: D significa deca (más comúnmente abreviado como da).

1349 Longitud

ID 1349 · título localizado en HelpEs

m: metro *
 cm: centímetro (1/100 m)
 mm: milímetro (1/1000 m)
 km: kilómetro (1.000 m)
 au: unidad astronómica (149.597.870.700 m)
 lyr: año-luz (9,46052840488E15 m)
 pc: parsec (3,261563407982 lyr)
 Å: angstrom (10^{-10} m)
 fermi: fermi (10^{-15} m)
 yd: yarda (3 pies)
 ft: pie (12 pulgadas)
 in: pulgada (2,54 cm)
 mile: milla (5.280 pies)
 fath: braza (6 pies)
 ftUS: pies U.S. (1200/3937 m)
 miUS: milla U.S. (5.280 ftUS)

rod: vara (16,5 pies)
chain: cadena (66 pies)
nmi: milla náutica (1.852 m)
mil: (1/1000 pulg.)

* = Unidad base SI

1350 Área

ID 1350 · título localizado en HelpEs

m²: metro cuadrado
km²: kilómetro cuadrado
cm²: centímetro cuadrado
yd²: yarda cuadrada
ft²: pie cuadrado
in²: pulgada cuadrada
mile²: milla cuadrada
ha: hectárea (10.000 m²)
a: área (100 m²)
acre: acre (43.560 ft²)
b: barn (10E-28 m²)
miUS²: milla cuadrada de agrimensura de EE. UU.

1351 Volumen

ID 1351 · título localizado en HelpEs

m³: metro cúbico
cm³: centímetro cúbico (1E-6 m³)
l: litro (1.000 cm³)
ml: mililitro (1 cm³)
yd³: yarda cúbica
ft³: pie cúbico
in³: pulgada cúbica
ozfl: onza líquida estadounidense (1/128 galUS)
ozUK: onza líquida imperial (1/160 galUK)
qt: cuarto líquido estadounidense (1/4 galUS)
liqpt: pinta líquida estadounidense (1/8 galUS)
ptUK: pinta imperial (1/8 galUK)
galUS: galón líquido estadounidense (3,785411784 l)
galUK: galón imperial (4,54609 l)
tbsp: cucharada estadounidense (1/2 ozfl)
tsp: cucharadita estadounidense (1/6 ozfl)
bu: bushel imperial (≈ 2.219,36 in³)
pk: peck estadounidense (1/4 buUS)
bbl: barril de petróleo (42 galUS)
st: estéreo (1 m³)
buUS: bushel estadounidense (≈ 2.150,42 in³)
cu: taza estadounidense (1/16 galUS)
fbm: pie tablar (1/12 ft³)

1352 Time

ID 1352 · título localizado en HelpEs

s: segundo *
min: minuto (60 s)
h: hora (60 min.)
d: día (24 h)
yr: año (≈ 365,2422 días)
Hz: hertz (ciclos por segundo)

* = unidad básica del SI

1353 Velocidad

ID 1353 · título localizado en HelpEs

m/s: metros por segundo
cm/s: centímetros por segundo
ft/s: pies por segundo
kph: kilómetros por hora
km/h: kilómetros por hora
mph: millas por hora (1,609344 km/h)
mile/h: millas por hora
knot: nudo (1 nmi/h o 1,852 km/h)
rad/s: radianes por segundo
tr/min: giros por minuto
tr/s: giros por segundo

1354 Masa

ID 1354 · título localizado en HelpEs

kg: kilogramo *
g: gramo (1/1000 kg)
slug: slug ($\approx 32,174$ lb)
lb: libra (0,45359237 kg)
oz: onza (1/16 lb)
tonUK: tonelada larga o tonelada imperial (2.240 lb)
tonUS: tonelada o tonelada corta (2.000 lb)
ozt: onza troy (31,1034768 g)
grain: grano (1/7.000 lb)
lbt: libra troy ($\approx 373,24$ g)
u: unidad de masa atómica unificada (1,660538921E-27 kg)
t: tonelada o tonelada métrica (1.000 kg)
ct: quilate (0,2 g)
mol: mol (6,0221412927E23) *

* = unidad básica del SI

1355 Aceleración

ID 1355 · título localizado en HelpEs

m/s²: metros por segundo por segundo
ft/s²: pies por segundo por segundo
grav: aceleración de la gravedad ($\approx 32,174$ ft/s² o $\approx 9,81$ m/s²)
Gal: galileo (1 cm/s²)
rad/sec²: radianes por segundo por segundo

1356 Fuerza

ID 1356 · título localizado en HelpEs

kg*m/s²: kilogramo metro por segundo por segundo
N: newton (1 kg m/s²)
dyn: dina (1 g cm/s²)
lbf: libra-fuerza (4,4482216 N)
kip: kip (1.000 lbf)
gf: gramo-fuerza (9,80665 mN)
pdl: poundal (1 lb ft/s²)

1357 Energía

ID 1357 · título localizado en HelpEs

kg*m²/s²: (1 julio)
J: julio (1 kg m²/s²)

Wh: vatio-hora (3.600 J)
 kWh: kilovatio-hora (3,6E6 J)
 ft*lb_f: pie-libra de fuerza ($\approx 1,355818$ J)
 kcal: kilocaloría (4186,8 J)
 cal: caloría (4,1868 J)
 eV: electronvoltio (1,60217773E-19 J)
 MeV: megaelectronvoltio (1.000.000 eV)
 Btu: unidad térmica británica (1.055,05585262 J)
 erg: erg (1E-7 J)
 thermEC: therm(EC) (1,05506E+8 J)
 thermUS: therm(US) (1,054804E+8 J)
 thermUK: therm(UK) ($\approx 105.505.585,257$ J)
 toe: tonelada equivalente de petróleo (41.868.000.000 J)
 tec: tonelada equivalente de carbón (2,784E10 J)
 lep: litro equivalente de petróleo (35.788.320 J)
 boe: barril equivalente de petróleo (6,1178632E9 J)

1358 Potencia

ID 1358 · título localizado en HelpEs

kg*m²/s³: (1 vatio)
 W: vatio (1 kg m²/s³)
 MW: megavatio (1.000.000 W)
 hp: caballo de fuerza (745,699871582 W)
 ft*lb_f/s: pie-libra-fuerza por segundo (1,35581794833 W)

1359 Presión

ID 1359 · título localizado en HelpEs

kg/m*s²: (1 pascal)
 Pa: pascal (1 kg/m*s²)
 bar: bar (100.000 Pa)
 atm: atmósfera estándar (101.325 Pa)
 psi: libra-fuerza por pulgada cuadrada (6.894,75729317 Pa)
 torr: torr (133,322368421 Pa)
 mmHg: milímetro de mercurio (133,322368421 Pa)
 inHg: pulgada de mercurio (3.386,38815789 Pa)
 inH₂O: pulgada de agua (248,84 Pa @ 60 °F)

1360 Temperatura

ID 1360 · título localizado en HelpEs

°C: escala Celsius
 (Punto de congelación del agua 0 °C)
 °F: escala Fahrenheit
 (Punto de congelación del agua 32 °F)
 K: escala Kelvin *
 (Punto de congelación del agua 273,15 K)
 °R: escala Rankine
 (Punto de congelación del agua 491,67 °R)
 * = unidad básica del SI

1361 Eléctrica

ID 1361 · título localizado en HelpEs

A: amperio (corriente eléctrica) *
 V: voltio (potencial eléctrico)
 C: culombio (carga eléctrica)

Ohm: ohmio Ω (resistencia)
F: faradio (capacitancia)
Fdy: faraday (96.485,3365 C)
Wb: weber (flujo magnético)
H: henry (inductancia)
mho: mho U (conductancia)
S: siemens (conductancia)
T: tesla (densidad de flujo magnético)
A*h: amperio-hora (3.600 C)
* = unidad básica del SI

1362 Luz

ID 1362 · título localizado en HelpEs

cd: candela (lm/sr) *
flam: foot-lambert ($\approx 3,426259 \text{ cd/m}^2$)
* = unidad básica del SI

1363 Ángulo

ID 1363 · título localizado en HelpEs

rad: radián ($180/\pi$ grados)
deg: grado ($\pi/180$ radianes)
grad: grado centesimal (1/400 de giro)
gon: gon (1/400 de giro)
arcmin: minuto de arco (1/60 de grado)
arcs: segundo de arco (1/3600 de grado)
tr: giro (360 grados)

1364 Viscosidad

ID 1364 · título localizado en HelpEs

P: poise, viscosidad dinámica ($0,1 \text{ kg}/(\text{m}\cdot\text{s})$)
 $\text{kg}/(\text{m}\cdot\text{s})$: unidad SI de viscosidad dinámica
 $\text{lb}/(\text{ft}\cdot\text{h})$: unidad imperial de viscosidad dinámica
St: stoke, viscosidad cinemática ($1 \text{ cm}^2/\text{s}$)
 m^2/s : unidad SI de viscosidad cinemática
 ft^2/s : unidad imperial de viscosidad cinemática

1365 Radiación

ID 1365 · título localizado en HelpEs

Actividad

Bq: becquerel (1 s^{-1})
Ci: curie ($3,7 \times 10^{10} \text{ Bq}$)

Dosis absorbida

Gy: gray (1 J/kg)
rd: rad ($0,01 \text{ Gy} = 0,01 \text{ J/kg}$)

Dosis equivalente

rem: equivalente de roentgen en el ser humano ($0,01 \text{ Sv}$)
Sv: sievert (1 J/kg , ponderado por radiación)

Exposición

R: roentgen ($2,58\text{E-}4 \text{ C/kg}$)

1366 Par de torsión

ID 1366 · título localizado en HelpEs

N*m: newton metro
kg*m²/s²: la versión del SI del newton metro
dyn*cm: dina centímetro
lbf*ft: libra-fuerza pie
lbf*inch: libra-fuerza pulgada
ozf*inch: onza-fuerza pulgada

1367 Constantes

ID 1367 · título localizado en HelpEs

Hay 29 constantes físicas de los campos de las matemáticas, la química, la física y la mecánica cuántica que puede usar en cálculos. Presione Shift Units y toque [[Const]] para mostrar un menú con estas constantes.

Toque [[Valor•]] para recuperar el valor de una constante con sus unidades.

Toque [[OK]] o presione Enter para copiar la constante en la línea de entrada.

1368 Matem.

ID 1368 · título localizado en HelpEs

Número de Euler

$e = 2,71828182846$

Número real máximo que HP Prime es capaz de representar

MAXREAL=1,79769313486E308 (Vista CAS)

MAXREAL=9,99999999999E499 (Vista de Inicio)

Número real mínimo (más cercano a cero) que HP Prime es capaz de representar

MINREAL=2,22507385851E-308 (Vista CAS)

MINREAL=1E-499 (Vista de Inicio)

pi

$\pi = 3,14159265359$

Valor imaginario

$i = \sqrt{-1}$

1369 Química

ID 1369 · título localizado en HelpEs

NA: Avogadro (6,02214129E23 1/mol)

k: Boltzmann (1,3806488E-23 J/K)

Vm: volumen molar del gas ideal (22,413968 l/mol)

R: constante molar de los gases (8,3144621 J/(mol K))

StdT: temperatura estándar (273,15 K)

StdP: presión estándar (101,325 kPa)

1370 Física

ID 1370 · título localizado en HelpEs

σ : Stefan-Boltzmann (5,670373E-8 W/(m² K⁴))

c: velocidad de la luz (299.792.458 m/s)

ϵ_0 : permitividad del vacío (8,8541878176E-12 F/m)

μ_0 : permeabilidad del vacío (1,25663706144E-6 H/m)

g: aceleración de la gravedad (9,80665 m/s²)

G: constante de gravitación universal (6,67384E-11 m³/(s² kg))

1371 Quantum

ID 1371 · título localizado en HelpEs

h: constante de Planck (6,62606957E-34 J s)

$\hbar$: constante de Dirac (1,054571729E-34 J s)

q: carga elemental (1,602176565E-19 C)

me: masa del electrón ($9,10938291\text{E-}31$ kg)
qme: relación q/me ($175.882.008.800$ C/kg)
mp: masa del protón ($1,672621777\text{E-}27$ kg)
mpme: relación mp/me ($1.836,15267245$)
 α : constante de estructura fina ($7,2973525698\text{E-}3$)
 Φ : cuanto de flujo magnético ($2,067833758\text{E-}15$ Wb)
F: constante de Faraday ($96.485,3365$ C/mol)
 R_∞ : constante de Rydberg ($10.973.731,5685$ 1/m)
 a_0 : radio de Bohr ($5,2917721092\text{E-}11$ m)
 μ_B : magnetón de Bohr ($9,27400968\text{E-}24$ J/T)
 μ_N : magnetón nuclear ($5,05078353\text{E-}27$ J/T)
 λ_0 : longitud de onda del fotón ($1,2398419292\text{E-}6$ m)
 f_0 : frecuencia del fotón ($2,41798934961\text{E}14$ Hz)
 λ_c : longitud de onda de Compton ($2,4263102389\text{E-}12$ m)

1372 Variables

ID 1372 · título localizado en HelpEs

Las variables son referencias a objetos con nombre (como definiciones de funciones, números, matrices, los resultados de los cálculos, etc.). Algunas están integradas y no se pueden eliminar. También puede crear sus propias variables.

Si se utiliza el nombre de una variable en una fórmula se devolverá el contenido (o el valor) de la variable.

Si se usa un nombre de variable como destino en una operación de STO (o :=) se altera su contenido (o valor).

Presione la tecla Vars para acceder a los menús de variables. Aparecerán los siguientes menús:

- [[Inicio]]
- [[CAS]]
- [[Apl.]]
- [[Usua.]]
- [[Catálogo]]

El menú de [[CAS]] se mostrará solo si ha creado sus propias variables en la vista de este sistema o si ha creado objetos en la aplicación Geometría.

El menú [[Usua.]] se mostrará solo si ha creado sus propias variables en la vista de Inicio o si ha creado variables globales en un programa.

El menú [[Catálogo]] se abre en un menú integrado por todas las variables definidas actualmente en su HP Prime.

Pulse la tecla del menú [[OK]] para copiar el nombre de la variable seleccionada en la línea de entrada o presione la tecla Esc para retroceder al menú Vars. En un menú de opciones, también puede seleccionar una entrada por su número o escribiendo la primera letra o dos de su nombre y presionando la tecla Enter.

1373 Variables de Inicio

ID 1373 · título localizado en HelpEs

El menú de variables de Inicio incluye todas las variables a las que se accede normalmente desde la vista Inicio o que afectan al funcionamiento de la vista de Inicio.

1374 Variables reales

ID 1374 · título localizado en HelpEs

Este menú contiene los nombres de las variables reales: A a Z y θ .

1375 Variables complejas

ID 1375 · título localizado en HelpEs

Este menú contiene los nombres de diez variables complejas: Z0 a Z9.

1376 Variables de lista

ID 1376 · título localizado en HelpEs

Este menú contiene los nombres de diez variables de lista: L0 a L9.

1377 Variables de matriz

ID 1377 · título localizado en HelpEs

Este menú contiene los nombres de diez matrices: M0 a M9.

1378 Variables de gráficos

ID 1378 · título localizado en HelpEs

Este menú contiene los nombres de diez variables de gráficos: G0 a G9.

1379 Variables de la configuración de Inicio

ID 1379 · título localizado en HelpEs

Este menú contiene los nombres de las variables utilizadas para controlar la configuración de Inicio. Son las siguientes:

- HAngle
- HFormat
- HDigits
- HComplex
- Entry
- Integer: Base
- Integer: Bits
- Integer: Signed

1380 HAngle — La variable de Inicio HAngle se utiliza para establecer el modo del ángulo para la vista de Inicio.

ID 1380 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: HAngle

Explicación española: La variable de Inicio HAngle se utiliza para establecer el modo del ángulo para la vista de Inicio.

La variable de Inicio HAngle se utiliza para establecer el modo del ángulo para la vista de Inicio.

HAngle := 0 para radianes (predeterminado)

HAngle := 1 para grados

HAngle := 2 para gradianes

1381 HFormat — La variable de Inicio HFormat controla el formato de visualización numérica utilizado en la vista de...

ID 1381 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: HFormat

Explicación española: La variable de Inicio HFormat controla el formato de visualización numérica utilizado en la vista de...

La variable de Inicio HFormat controla el formato de visualización numérica utilizado en la vista de Inicio. Esta variable puede contener cualquier número entero entre 0 y 5, cada valor tiene el siguiente significado:

HFormat :=0 para Estándar

HFormat :=1 para Fijo

HFormat :=2 para Científico

HFormat :=3 para Ingeniería

HFormat :=4 para Flotante

HFormat :=5 para Redondeo

1382 HDigits — La variable de Inicio HDigits controla la cantidad de dígitos mostrados después del punto decimal cuando...

ID 1382 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: HDigits

Explicación española: La variable de Inicio HDigits controla la cantidad de dígitos mostrados después del punto decimal cuando...

La variable de Inicio HDigits controla la cantidad de dígitos mostrados después del punto decimal cuando el formato numérico no es el Estándar.

HDigits := n, donde n es un entero de forma que $0 \leq n \leq 11$.

1383 HSeparator — La variable de Inicio HSeparator se utiliza para controlar los separadores usados en exhibición del...

ID 1383 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: HSeparator

Explicación española: La variable de Inicio HSeparator se utiliza para controlar los separadores usados en exhibición del...

La variable de Inicio HSeparator se utiliza para controlar los separadores usados en exhibición del número.

HSeparator puede contener cualquier número entero de 0 a 10 correspondiente a la agrupación de dígitos seleccionada en la página 1 de la Configuración de inicio.

1384 HComplex — La variable de Inicio HComplex se utiliza para controlar la configuración Compleja.

ID 1384 · identificador de HelpEn con explicación española de HelpEs

Referencia traducida

Identificador técnico: HComplex

Explicación española: La variable de Inicio HComplex se utiliza para controlar la configuración Compleja.

La variable de Inicio HComplex se utiliza para controlar la configuración Compleja.

Esta variable contiene un número entero de dos dígitos. El dígito de las unidades de HComplex controla el resultado complejo de la entrada real (1: activado, 0: desactivado)

El dígito de las decenas controla la visualización de los números complejos (0: $a+bi$, 1: (a,b))

Thus HComplex := 10; establece la presentación de números complejos como (a,b) y resultados complejos de bloque a partir de entradas reales.

Con el dígito de las unidades de HComplex ajustado a 0, ASIN(2) devuelve un error; ajustado a 1, ASIN(2) devuelve $1.57079632679-1.31695789692*i$.

1385 Variable de entrada

ID 1385 · título recuperado de la página española

Devuelve o establece el modo de entrada.

Entry := 0 para libro de texto (predeterminado)

Entry := 1 para algebraico

Entry := 2 para RPN

1386 Variable Base

ID 1386 · título recuperado de la página española

Devuelve o establece el formato de base de entero.

Base := 0 para binario

Base := 1 para octal

Base := 2 para decimal

Base := 3 para hexadecimal (predeterminado)

1387 Signed Variable

ID 1387 · título recuperado de la página española

Devuelve o establece el formato del signo del entero.

Signed := 0 sin signo (predeterminado)

Signed := 1 con signo

1388 Variable Bits

ID 1388 · título recuperado de la página española

Devuelve o establece el tamaño de bits de enteros.

Bits := x, donde $1 \leq x \leq 64$

1389 Variables del sistema

ID 1389 · título localizado en HelpEs

Este menú muestra los nombres de las variables utilizadas para controlar los ajustes o los objetos del sistema. Ellas son:

- Date (Fecha)
- Time (Hora)
- language (Idioma)
- Notes (Notas)
- Programs (Programas)
- TOff
- TDim
- Theme
- Apps
- DelApp
- HVars
- DelHVars

1390 Variables de fecha

ID 1390 · título recuperado de la página española

Devuelve la fecha del sistema en el formato AAAA.MMDD.

Date:= AAAA.MMDD cambia la fecha.

1391 Variable System Time

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Variable System Time.](#)

1392 Variable Idioma

ID 1392 · título recuperado de la página española

La variable de Inicio Language controla qué idioma se utiliza en la vista de Inicio.

Language:= 1 para inglés (opción predeterminada)

Language:= 2 para chino

Language:= 3 para francés

Language:= 4 para alemán

Language:= 5 para español

Language:= 6 para holandés

Language:= 7 para portugués

Language:= 8 para japonés

1393 Variable Notas

ID 1393 · título recuperado de la página española

La variable Notes proporciona acceso a las notas guardadas en la calculadora.

Sin argumento, Notes devuelve una lista de los nombres de todas las notas en la calculadora.

Notes(n) devuelve el contenido de la enésima nota en la calculadora (1 a número de notas).

Notes("name") devuelve el contenido de la nota llamada con ese nombre.

Este comando también se puede utilizar para definir, redefinir o borrar una nota.

Notes(n):="string" establece el valor de la nota n. Si la cadena está vacía, la nota se borra.

De forma similar, Notes("name"):= "string" establece el valor de la nota "name". Si la cadena está vacía, la nota se borra. Si no hay nota llamada "name", la crea con una cadena como contenido.

1394 Variable Programas

Sintaxis

Programs
 Programs(n)
 Programs(name)
 Programs(n):=source
 Programs(name):=source

La variable Programs proporciona acceso a los programas guardados en la calculadora.

Parámetros

- n: Índice de un programa, de 1 al número de programas almacenados en la calculadora.
- name: Nombre de un programa, especificado como una cadena.
- source: Código fuente del programa, especificado como una cadena.

Programs devuelve una lista que contiene los nombres de todos los programas almacenados en la calculadora.

Programs(n) y Programs(name) devuelven el contenido del programa especificado.

Si el programa consta de un único archivo de código fuente, su contenido se devuelve directamente como una cadena.

Para los programas con contenido más complejo, se devuelve una lista que describe el contenido del programa. Esta puede incluir varios archivos de código fuente y otros recursos del programa, junto con información sobre sus tipos.

Programs(n):=source establece el código fuente del programa n. Si source es una cadena vacía, se elimina el programa.

Programs(name):=source establece el código fuente del programa llamado name. Si source es una cadena vacía, se elimina el programa. Si no existe ningún programa con ese nombre, se crea uno nuevo.

1395 Variable Time Off

TOff contiene un entero que define la cantidad de milisegundos hasta la próxima vez que la calculadora se apague. El tiempo predeterminado es de 5 minutos, o #493E0h. (5*60*1000 milisegundos)

Los rangos válidos son de #1388h a #3FFFFFFh.

1726 Variable de tiempo de atenuación

La variable TDim contiene un entero que define el número de milisegundos antes de que la pantalla de la calculadora se oscurezca automáticamente. El valor predeterminado es de 30 segundos, o #7530h (30 * 1000 milisegundos).

Los rangos válidos son de #1388h a #3FFFFFFh. En algunas calculadoras, este valor puede truncarse al entero más cercano.

Tenga en cuenta que establecer este valor más alto que el predeterminado afectará negativamente la vida útil de la batería de la calculadora.

1396 Variable Theme

La variable Theme contiene una lista que representa el tema actual y la tonalidad de color, tal como se define en la página 2 de la Configuración de inicio. Puede tomar un solo número como entrada para devolver el elemento indexado de la lista.

0066 Lista de aplicaciones**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Lista de aplicaciones.](#)

0067 Eliminar aplicación**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Eliminar aplicación.](#)

1397 Variables de Inicio

ID 1397 · título recuperado de la página española

Permite el acceso a las variables de inicio definidas por el usuario.

HVars devuelve una lista de los nombres de todas las variables de inicio definidas por el usuario.

HVars(n) devuelve la n-ésima variable de inicio definida por el usuario.

HVars("name") devuelve la variable de inicio definida por el usuario con el nombre dado.

HVars(n or "name", 2) si la variable es una función de usuario, devuelve la lista de los parámetros para esa función; de lo contrario devuelve 0.

HVars(n):=value almacena el valor en la enésima variable de inicio definida por el usuario.

HVars("name"):={object} almacena un objeto en la variable de inicio definida por el usuario llamada "name". Si no existe dicha variable, la crea.

HVars(n o "nombre", 2):= {"Param1Name", ..., "ParamNName"} supone que la variable de usuario especificada contiene una función. Especifica cuáles son los parámetros de esa función.

HVars("name", {"param",...}):='función' crea una función de usuario llamada nombre con parámetros como entradas y la función se define conforme a esas entradas.

Ejemplos

[Ejemplo gráfico E1] crea una función de usuario llamada Actual para usar la Ley de Ohm para calcular el valor de I (actual) dadas las variables de entrada V (voltaje) y R (resistencia).

1398 Eliminar las variables de inicio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Eliminar las variables de inicio.](#)

1399 Variables del sistema algebraico computacional (CAS, por sus siglas en inglés)

ID 1399 · título recuperado de la página española

El menú CAS aparece solo si creó sus propias variables en la Vista CAS o si creó objetos en la aplicación Geometría. A los objetos creados en la aplicación Geometría se les da un prefijo "G" (tales como GA, GB, etc.).

1400 Variables de aplicación

ID 1400 · título localizado en HelpEs

Las variables de aplicación son variables para configuraciones y resultados de una aplicación específica. Las variables se ordenan por aplicación.

En el caso de aplicaciones de producen resultados, cada variable RESULTS se puede usar en una expresión. Por ejemplo, la aplicación Función le permite calcular varios puntos críticos, como las intersecciones, las raíces, la extrema y la probabilidad. Si, por ejemplo, ha usado la aplicación Función para calcular las raíces de una función, podría seleccionar la variable Root y presionar Enter. Se recupera el valor de la variable. También podría incluir Root en una expresión como $X^2 - 2X + \text{Root}$.

Tenga en cuenta que los nombres de variables distinguen entre mayúsculas y minúsculas.

1401 Variables de usuario

ID 1401 · título localizado en HelpEs

El menú Usua. se mostrará solo si ha creado sus propias variables en la vista de Inicio; o bien, si ha creado variables globales en un programa utilizando el comando EXPORT.

1681 Catálogo Vars

El catálogo Vars contiene una lista completa de todas las variables definidas actualmente en su HP Prime.

Al lado derecho del encabezado del catálogo Vars hay un icono de información pequeño (i). Pulse el icono para ver el número de cada tipo de variable definida actualmente en su HP Prime.

1000 Variables de PPL

En HP PPL, las variables se utilizan para almacenar valores durante la ejecución del programa. Las variables no tienen tipado estático, lo que significa que una variable no tiene un tipo de datos fijo asignado. En cambio, el tipo depende del valor almacenado actualmente en la variable.

Se puede asignar un valor a una variable utilizando el operador de asignación := o el operador de almacenamiento ►. Cuando se asigna un nuevo valor, reemplaza el contenido anterior de la variable.

Dependiendo de dónde y cómo se declaren, las variables pueden ser locales a un programa, exportadas a nivel de programa o locales a una función. Esto determina el ámbito y la vida útil de la variable dentro del programa.

En HP PPL, una variable declarada fuera de cualquier cuerpo de función es una variable de nivel de programa. Es global dentro del programa.

Si las variables de nivel de programa se exportan, también se puede acceder a ellas desde fuera del programa en el que se declararon.

```
// variable exportada de nivel de programa
EXPORT counter:=0;

// variable local de nivel de programa
LOCAL cache:={ };
temp:=5; // implícitamente local al programa
```

Una variable local a una función se declara con LOCAL dentro del cuerpo de una función. Existe solo mientras se está ejecutando la función o el bloque en el que fue declarada.

```
EXPORT demo()
BEGIN
// variable local a la función
LOCAL x:=0;
END;
```

Las variables de nivel de programa, ya sean exportadas o locales al programa, permanecen en memoria después de que el programa finaliza. Sus valores no se restablecen automáticamente cada vez que se ejecuta el programa.

Una variable de nivel de programa se inicializa cuando se compila el programa. No se reinicializa automáticamente en cada ejecución. Por lo tanto, si su valor cambia durante la ejecución, el valor modificado puede seguir estando presente la próxima vez que se ejecute el programa.

Ejemplo

```
counter:=0;

EXPORT demo()
BEGIN
counter:=counter+1;
RETURN counter;
END;
```

Si el programa se ejecuta repetidamente sin recompilación, counter puede seguir aumentando en lugar de restablecerse a 0 en cada ejecución.

Para restablecer una variable cada vez que se ejecuta el programa, asigne su valor inicial explícitamente dentro del cuerpo de la función:

```
counter;

EXPORT demo()
BEGIN
counter:=0;
...
END;
```

Las variables declaradas dentro de una función se inicializan cuando la ejecución alcanza la declaración, por lo que se recrean e inicializan cada vez que se ejecuta ese bloque.

Ejemplo

```
EXPORT demo()  
BEGIN  
  LOCAL counter:=0;  
  counter:=counter+1;  
  RETURN counter;  
END;
```

En este caso, counter es una variable local a la función y se inicializa a 0 cada vez que se llama a demo().

Debido a que las variables de nivel de programa permanecen en memoria, pueden seguir ocupando memoria después de que el programa finaliza. Esto es especialmente importante cuando contienen listas grandes, matrices, cadenas u otras estructuras de datos grandes.

Si dichos datos ya no son necesarios, considere limpiar la variable antes de que el programa termine, por ejemplo:

```
largeData:=0;
```

Esto no elimina la declaración de la variable, pero libera el valor grande almacenado en la variable.

1402 Gestor de memoria

ID 1402 · título localizado en HelpEs

La tecla Shift del cuadro de herramientas (Mem) le lleva al Gestor de memoria, que es básicamente un catálogo de catálogos. Aquí aparecen todos los catálogos. Pulse el botón de menú `[[Ver]]`, o presione Enter, para acceder a un catálogo seleccionado y eliminar los objetos que ya no necesita.

Botones de menú:

- `[[Info]]` muestra el espacio de almacenamiento y memoria
- `[[Clonar]]` clona la calculadora HP Prime actual a la calculadora HP Prime conectada
- `[[Enviar]]` envía los datos de categorías determinadas (Listas, Matrices, etc.) a una calculadora HP Prime conectada
- `[[Vista]]` abre el catálogo seleccionado

1636 Copias de seguridad

ID 1636 · título localizado en HelpEs

Esta pantalla le permite guardar, restaurar o crear nuevas copias de seguridad completas de su calculadora.

Botones de menú

- `[[Restaurar]]` sustituye el contenido actual de su calculadora por los archivos de copias de seguridad seleccionados
- `[[Eliminar]]` elimina el archivo de copia de seguridad seleccionado actualmente
- `[[Nuevo]]` abre un diálogo para crear una nueva copia de seguridad. De forma predeterminada, se brindan la palabra "Copia de seguridad" y la fecha actual.

1633 Hacer copia de seguridad de la memoria de la calculadora

ID 1633 · título recuperado de la página española

Sintaxis

`Backup("nombre")`

Hace una copia de seguridad de toda la memoria de la calculadora bajo "nombre".

1634 Restarar la memoria de la calculadora

ID 1634 · título recuperado de la página española

Sintaxis

`Restore("nombre")`

Ejemplos

[Ejemplo gráfico E1] Devuelve la lista de todas las copias de seguridad disponibles.

[Ejemplo gráfico E2] Restaura la memoria de la calculadora al estado guardado bajo "nombre". La memoria actual de la calculadora se borrará.

1635 Eliminar las copias de seguridad de la calculadora

ID 1635 · título recuperado de la página española

Sintaxis

`DelBackup("nombre")`

Elimina la copia de seguridad "nombre".

1403 Menú de caracteres

ID 1403 · título localizado en HelpEs

Si presiona Shift y la tecla Vars (Chars) llega al menú de Caracteres que contiene el conjunto de caracteres ampliado de la calculadora HP Prime. Utilice la rueda para seleccionar un carácter del conjunto actual. Pulse dos veces sobre el carácter, o pulse [[Eco]] para agregarlo a la cadena actual en el área de entrada. Seleccione tantos caracteres como desee. Pulse [[OK]] para introducirlos en la ubicación del cursor actual o presione Esc para deshacerse de los caracteres y volver a la línea de edición.

Botones de menú:

- [[Eco]] añade el carácter seleccionado a una cadena de uno o más caracteres para ser insertado en la línea de edición actual
- [[Más]] permite elegir entre una selección de conjuntos de caracteres, cada uno de los cuales contiene varias páginas
- [-▲ Página ▼-] permite ir hacia delante o hacia atrás por las distintas páginas del conjunto de caracteres actual
- [[OK]] cierra el menú de caracteres y pega la cadena de caracteres actual en la línea de edición

1404 Paletas de atajo

ID 1404 · título localizado en HelpEs

Muchas operaciones y caracteres pueden introducirse simplemente seleccionándolos desde las paletas. Existen:

- Paleta de plantilla matemática: la tecla de plantilla matemática muestra una paleta de plantillas preformateadas que representan operaciones y expresiones matemáticas comunes, tales como raíz n -ésima, diferenciación e integración (además de vectores, matrices y números sexagesimales).
- Paleta de relaciones - (Shift 6) muestra operadores booleanos y relacionales útiles en matemáticas y programación (tales como mayor que, no igual a, y OR).
- Paleta de caracteres especiales - (Shift 9) muestra una paleta de caracteres comunes en matemáticas y estadísticas (como infinito, promedio y desviación estándar), así como caracteres griegos

1405 Plantilla de matemáticas

ID 1405 · título localizado en HelpEs

La tecla de plantilla matemática muestra una paleta de plantillas preformateadas que representan operaciones y expresiones matemáticas comunes, tales como raíz n -ésima, diferenciación e integración (además de vectores, matrices y números sexagesimales).

Toque la plantilla que desea o utilice la rueda para resaltar la plantilla y presione Enter. La plantilla se copia en la línea de entrada y permite agregar los valores.

Presione Esc para cerrar la paleta sin realizar una selección.

1406 Plantilla de matemáticas

ID 1406 · título localizado en HelpEs

La tecla de plantilla matemática muestra una paleta de plantillas preformateadas que representan operaciones y expresiones matemáticas comunes, tales como raíz n -ésima, diferenciación e integración (además de vectores, matrices y números sexagesimales).

Toque la plantilla que desea o utilice la rueda para resaltar la plantilla y presione Enter. La plantilla se copia en la línea de entrada y permite agregar los valores.

Presione Esc para cerrar la paleta sin realizar una selección.

1407 Paleta de caracteres especiales

ID 1407 · título localizado en HelpEs

Presione Shift 9 (!, ∞ , $\rightarrow$) para mostrar la paleta de caracteres especiales. Esta paleta muestra muchos símbolos matemáticos comunes (como el infinito, promedio y la desviación estándar), así como caracteres griegos.

Presione Esc para cerrar la paleta sin realizar una selección.

Tenga en cuenta que puede mostrar una lista completa de caracteres pulsando Shift + Vars (Chars).

1408 Paleta de relaciones

ID 1408 · título localizado en HelpEs

Presione Shift 6 ($\leq$, $\geq$, $\neq$) para mostrar la paleta de relaciones. Esta paleta muestra operadores booleanos y relacionales útiles en matemáticas y programación (como $\leq$, $=$, $>$, AND, y OR).

Presione Esc para cerrar la paleta sin realizar una selección.

1409 Teclado

ID 1409 · título localizado en HelpEs

Aquí puede conseguir ayuda sobre cualquier tecla del teclado y cualquier combinación de la tecla Shift.

1410 Conversiones Decimales: a_b/c

ID 1410 · título recuperado de la página española

Conversión de decimales a fracciones

En vista de Inicio, alterna la última entrada entre decimal, fracción y las formas de números mixtos. Si se selecciona un resultado del historial, entonces alterna la selección a través de estas formas. También funciona con listas y matrices.

En la vista CAS, solo alterna entre decimales y fracciones equivalentes, y los agrega como nuevas entradas al Historial.

1411 Almohadilla direccional

ID 1411 · título localizado en HelpEs

La almohadilla direccional mueve el cursor en la pantalla.

Presione primero la tecla Shift para ir al principio, el fin, la parte superior, o la parte inferior de una lista o de la pantalla.

Presione la tecla ALPHA primero para pasar 1 pantalla a la vez.

En la Vista de gráfico, con TRAZAR activado, la rueda hacia la izquierda/derecha mueve el trazador a la izquierda y derecha del gráfico actual. Utilice la rueda hacia arriba/abajo para cambiar al gráfico activo anterior o siguiente.

1412 Tecla X,T,θ,N

ID 1412 · título recuperado de la página española

La tecla X,T,θ,N es una ayuda de escritura que inserta la variable independiente apropiada para la aplicación actual.

En la aplicación Paramétrica introduce T.

En la aplicación Polar introduce θ.

En la aplicación Secuencia introduce N.

En la aplicación Soluc, introduce la primera variable que se encuentra en la ecuación marcada.

En todas las otras aplicaciones, introduce X.

1413 Retroceso

ID 1413 · título localizado en HelpEs

La tecla retroceso borra el carácter a la izquierda del cursor.

En la pantalla de configuración, como Configuración de gráfico, presione esta tecla para restablecer cualquier campo a su valor predeterminado.

En el historial, presione esta tecla para eliminar la entrada o resultado resaltados.

1414 Tangente

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tangente.](#)

1415 Función de cuadrado

ID 1415 · título recuperado de la página española

Sintaxis Valor^2

Devuelve el cuadrado de Valor.

1416 Función de potencia

ID 1416 · título recuperado de la página española

Sintaxis x^y

Devuelve el resultado de elevar x a la y potencia. También acepta números complejos.

1417 Dividir**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dividir.](#)

1418 Coma

ID 1418 · título recuperado de la página española

Ingresa en un carácter separador. El carácter introducido depende de la agrupación de dígitos seleccionados en la página 1 de la Configuración de Inicio.

1419 Numerales

ID 1419 · título localizado en HelpEs

Ingresa los números del 0 al 9.

1420 Multiplicar**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Multiplicar.](#)

1421 Restar**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Restar.](#)

1422 Agreg.**Referencia cruzada**

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Agreg..](#)

1423 Punto decimal

ID 1423 · título recuperado de la página española

Si presiona la tecla de punto (.) decimal se escribe el carácter de la marca decimal. El carácter introducido depende del Grupo de dígitos seleccionados en la página 1 de Configuración de Inicio.

Si el Grupo de dígito seleccionado utiliza la coma como la marca decimal, presione ALPHA antes de presionar esta tecla para ingresar un punto.

1424 Tecla Enter

Ejecuta la expresión presente en la línea de entrada. También funciona como el botón de menú **[[OK]]** para aceptar el estado actual de un campo de entrada.

1425 Crear una función definida por el usuario

El cuadro de diálogo Defn le permite definir una función de usuario sin crear un programa. Solo tiene que introducir el nombre de la función en el campo Nombre y el tipo de función en el campo Función.

Al tocar **[[OK]]**, se mostrará una lista de las variables utilizadas en la función. Compruebe todas las variables que sean ENTRADAS en su función, pulse **[[OK]]** y habrá terminado. Ya puede utilizar la función en el sistema.

Ejemplo: para crear la función $SINCOS(A,B)=SIN(A)+COS(B)+C$:

Introduzca SINCOS como el nombre, $SIN(A)+COS(B)+C$ como la función, toque **[[OK]]**, compruebe A y B, y anule la selección de C. Ahora toque **[[OK]]**.

A continuación, en la línea de entrada, introduzca $SINCOS(1,2)$ para calcular $SIN(1)+COS(2)+C$.

Si introduce el nombre de una función existente en el campo Nombre, el campo Función se completará automáticamente con la función asociada a dicho nombre. Esto le permitirá modificar fácilmente la definición.

En el cuadro de diálogo Definir, pulse **[[Elegir]]** para ver una lista de todas las funciones definidas por el usuario. Para borrar una función, utilice la rueda para resaltarla en la lista y luego presione la tecla de retroceso. Para editar una función, simplemente selecciónela en la lista.

1426 Grados Minutos Segundos

Conversión decimal a sexagesimal

Esta combinación de tecla Shift cambia un resultado numérico entre las representaciones decimales y sexagesimales.

Cualquier resultado decimal puede mostrarse como sexagesimal; es decir, en unidades subdivididas en grupos de 60. Esto incluye grados, minutos y segundos y también horas, minutos y segundos. Introduzca su expresión en la Vista de Inicio y luego presione la tecla Shift, seguido de esta tecla, para convertir el resultado a sexagesimal. Repita el procedimiento para volver a una representación decimal.

Durante el ingreso de cualquier expresión, esta combinación de tecla Shift inserta los símbolos de grados, minutos y segundos ($^{\circ}$, $'$, $''$).

1427 Tecla EEX

La tecla EEX se utiliza para introducir números en notación 'exponencial', también conocida como notación científica, forma científica, forma estándar o forma de índice estándar.

En el caso de HP Prime, un número en la notación exponencial está representado por 2 partes separadas por el carácter ϵ (correspondiente a la tecla EEX).

La primera parte suele llamarse la mantisa y es un número real.

La segunda parte suele llamarse el exponente y es un entero.

El número representado por esta notación es $\text{mantisa} \times 10^{\text{exponente}}$.

Por lo tanto, al escribir 3 EEX 5 se genera $3\epsilon 5$ en la línea de comando, que devuelve el número 300000.

1428 Tecla Eliminar

La combinación de teclas Shift-Esc (Clear) borra la línea de edición, si está activa. En un campo en una visión, como en Configuración del gráfico, presione esta tecla para restaurar todos los campos en la vista a sus valores predeterminados. En el historial de la pantalla, presione esta tecla para borrar todas las entradas y resultados en el historial.

1429 Función exponencial

ID 1429 · título recuperado de la página española

Sintaxis

e^{Valor}
 $e^{(\text{Expr})}$

La combinación de las teclas Shift LN (e^x) es la función exponencial; devuelve e elevado a la potencia de Valor.

1430 Raíz cuadrada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Raíz cuadrada.](#)

1431 Copiar

ID 1431 · título recuperado de la página española

Pulse Shift View (Copy) para copiar el valor, expresión, texto u objeto resaltado en el portapapeles.

1432 Pegar

ID 1432 · título recuperado de la página española

Presione Shift Menu (Paste) para abrir el portapapeles de Paste.

Una vez abierto el portapapeles, puede desplazarse para resaltar un objeto. Pulse el menú **[[OK]]** para pegar el objeto resaltado en la línea de edición en la posición del cursor o en un campo seleccionado en un formulario de entrada.

Pulse la tecla Esc para cerrar el portapapeles sin pegar nada.

Botones de menú:

- **[[Mostr.]]** permite mostrar el elemento seleccionado a pantalla completo con el formato de libro de texto
- **[[Borrar]]** permite borrar todos los elementos del portapapeles
- **[[Elimi.]]** permite eliminar el elemento seleccionado del portapapeles
- **[[OK]]** permite pegar el elemento seleccionado en la línea de edición en la posición del cursor o en el campo seleccionado en un formulario de entrada

1433 Inverso

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Inverso.](#)

1434 Ángulo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ángulo.](#)

1435 Tecla Base

ID 1435 · título recuperado de la página española

La tecla Base abre la pantalla Editar entero. Si el elemento seleccionado actualmente (o el resultado más reciente) es un número entero, ese valor se carga en la pantalla para su edición.

En el cuadro de diálogo Editar entero, el campo Era en la parte superior muestra el entero original que seleccionó en la Vista de inicio. El campo Salida muestra el número entero editado. Ambos números enteros se muestran inicialmente en la base predeterminada, tal como se especifica en la página 1 de la Configuración de inicio.

El campo 16 es la representación hexadecimal de la Salida.

El campo 10 es la representación decimal de la Salida.

El cuadro debajo del valor decimal muestra la representación binaria (bit) de 64 bits de la Salida.

Cambiar el valor por cualquiera de los siguientes métodos actualiza el valor de la Salida y todas las representaciones:

- Desplace el número entero a la izquierda o a la derecha. Se pierden los bits desplazados en ambos extremos. Arrastre la pantalla hacia la izquierda o hacia la derecha para desplazarse 1 bit en la dirección correspondiente. Mueva la rueda a la izquierda/derecha para cambiar 1 bit. Mueva la rueda alfa a la derecha/izquierda para cambiar 4 bits (1 nibble [cantidad de datos de 4 bits]). Desplace la rueda a la derecha/izquierda 8 bits (un byte).
- Cambie el tamaño de la palabra (número de bits) del número entero. Arrastre la pantalla hacia abajo/arriba para aumentar/disminuir 1. Presione la rueda hacia arriba/abajo para aumentar/disminuir 1. Mueva la rueda alfa hacia arriba/abajo para aumentar/disminuir 4 bits (1 nibble [cantidad de datos de 4 bits]). Desplace la rueda hacia arriba/abajo para aumentar/disminuir 8 bits (1 byte).
- Cambie el signo del entero: Presione la tecla +/-
- Desplácese por los ajustes de base (sistema, hexadecimal, decimal, octal y binario): teclas + y -

Botones de menú:

- [[Rein.]] devuelve todos los cambios a su estado original
- [[Base]] pasa por las distintas bases; como ocurre al presionar +
- [[Firmada]] cambia el tamaño de las palabras entre con signo y sin signo
- [[NO]] devuelve el complemento de uno, (es decir, cada bit del tamaño de palabra especificado se invierte: 0 se sustituye por 1 y 1 por 0)
- [[Editar]] activa el modo de edición. El modo de edición está activo si aparece un carácter de viñeta en el botón [[Editar]]. Cuando el modo de edición está activo, un selector de dígitos resalta un solo dígito y puede desplazarse por el cuadro de diálogo con ayuda de la rueda. Los campos hex y decimal se pueden modificar, al igual que la representación de bits, un dígito por vez. Un cambio en un campo modifica automáticamente el resto de campos.
- [[OK]] cierra el cuadro de diálogo y guarda sus cambios. Si no desea guardar los cambios, presione Esc.

Valide el cambio en el número mediante [[OK]] o bien, cancele el cambio mediante la tecla Esc.

1436 Tecla igual

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tecla igual.](#)

1437 Tecla CAS

ID 1437 · título recuperado de la página española

Abre el Sistema algebraico computacional (CAS).

1438 Tecla Vars

ID 1438 · título recuperado de la página española

La tecla Vars abre los menús para que usted elija las variables. Puede seleccionar variables de Inicio, CAS, Apl. y Usuar. desde estos menús.

1439 Tecla Apps

ID 1439 · título recuperado de la página española

Si presiona Apps abre la Biblioteca de aplicaciones, en la que puede seleccionar una aplicación para operaciones de restablecimiento, apertura o envío.

1440 Vista simbólica

ID 1440 · título recuperado de la página española

Tecla Symb

Abre la Vista simbólica para la aplicación activa. Lo que puede hacer en esta vista depende de la aplicación. Por ejemplo, puede definir funciones y sentencias abiertas, crear objetos geométricos, configurar una prueba de hipótesis y definir análisis estadísticos

Tenga en cuenta que la Vista simbólica no se utiliza en las aplicaciones Hoja de cálculo, Finanzas, Soluc. y Explorador.

1441 Vista de gráfico

ID 1441 · título recuperado de la página española

Tecla Plot

Abre la Vista de gráfico de la aplicación activa. Las acciones que realice en esta Vista dependen de la aplicación. Por ejemplo, puede trazar funciones y sentencias abiertas, crear objetos geométricos y explorar gráficos lineales, cuadráticos, sinusoidales y de amortización.

Tenga en cuenta que la Vista de gráfico no se utiliza en las aplicaciones Hoja de cálculo, Soluc. de triáng. o Soluc. lineal.

1442 Vista numérica

ID 1442 · título recuperado de la página española

Tecla Num

Abre la Vista numérica para la aplicación activa. Qué se puede hacer en esta vista depende de la aplicación. Por ejemplo, puede explorar tablas de valores generados por funciones, realizar mediciones geométricas, hacer cálculos de hoja de cálculo e ingresar datos para análisis estadísticos.

1443 Tecla de menú

ID 1443 · título recuperado de la página española

Tecla de menú

El menú con reconocimiento de contexto brinda opciones para que usted:

- Copie un elemento de la vista de Inicio a la vista del sistema algebraico computacional
- Copie un elemento de la vista del sistema algebraico computacional a la vista de Inicio
- Vea los mensajes que ha recibido a través del Kit de conectividad

La tecla de Menú también puede darle acceso a funciones para su aplicación actual. Por ejemplo, en la aplicación Hoja de cálculo, el menú incluye funciones como SUM, AVERAGE, AMORT, etc.

1444 Tecla Escape

ID 1444 · título recuperado de la página española

Esc: Escape

- Borra la línea de entrada o cierra un menú.
- Cierra un menú o una ventana emergente
- Cierra la mayoría de las vistas y los formularios de entrada
- Cancela los cambios en un campo de entrada que se está editando

1445 Paréntesis

ID 1445 · título recuperado de la página española

Pulse () para insertar un par de paréntesis.

1446 Tecla ALPHA

ID 1446 · título recuperado de la página española

Presione la tecla ALPHA para acceder al carácter de texto impreso en naranja en la parte inferior derecha de una tecla. Por ejemplo, para escribir Z en la Vista de Inicio, presione ALPHA 2.

Cuando pulse ALPHA, se mostrará el anunciador A ... Z en la barra de título. Esto indica que la siguiente tecla que presione insertará texto en mayúsculas. Sin embargo, en la vista CAS, se mostrará el anunciador a ... z, indicando que la siguiente tecla insertará texto en minúsculas.

Presione Shift para cambiar el uso de mayúsculas y minúsculas: el anunciador cambiará para indicar el uso de mayúsculas y minúsculas del siguiente carácter.

Presione ALPHA ALPHA para poner el teclado en modo de bloqueo. Aquí puede escribir más de una letra de forma consecutiva. El anunciador de la barra de título cambia a AZ indicando el bloqueo en mayúsculas (o az en CAS).

Presione Shift antes de la tecla para cambiar el uso de mayúsculas y minúsculas; el anunciador cambia a a..z e indica que el siguiente carácter va en minúsculas (A..Z en CAS).

Presione Shift ALPHA para cambiar el uso de minúscula y mayúscula del bloqueo; el anunciador cambiará de AZ a az o de az a AZ.

Presione ALPHA para salir del bloqueo alfabético. El anunciador de texto se borra.

1447 Tecla Shift

ID 1447 · título recuperado de la página española

Presione Shift una vez para insertar un carácter de tecla de color azul.

Utilícela junto con ALPHA para introducir caracteres en minúsculas.

1448 Tecla On

ID 1448 · título recuperado de la página española

Enciende la calculadora.

Una vez activada, la tecla funciona como una tecla Esc alternativa. Puedo usarla para borrar la línea de entrada o cerrar un menú.

Mantenga presionada la tecla On y presione una segunda tecla para realizar las siguientes acciones.

- Borrar datos en la memoria : On & Apps & Esc
- Reiniciar : On & Symb
- Pantalla del terminal : On & T (+)
- Disminuir el brillo de la pantalla: On & -
- Aumentar el brillo de la pantalla : On & +
- Pantalla del modo de examen : On + Esc u On + E (a b/c)

1449 Tecla espaciadora

ID 1449 · título recuperado de la página española

Espacio: espacio alfabético

Introduce un espacio

1450 Tecla Sto►

ID 1450 · título recuperado de la página española

Almacena un valor en una variable (es decir, asigna un valor a una variable). Si después desea utilizar ese valor en un cálculo, puede hacer referencia al él por el nombre de variable.

Ejemplo: 3 ► A almacena el valor 3 en la variable A

Puede acceder al comando de almacenar presionando Shift EEX.

1451 Vista Config. simbólica

ID 1451 · título recuperado de la página española

Muestra la vista de la Config. simbólica, que es la misma para cada aplicación. Permite anular la configuración del sistema para

la medida del ángulo, el formato de número y la introducción de números complejos. El ajuste "Sistema" indica que se utilizará la configuración del sistema para ese campo.

La anulación se aplica a todas las funciones siempre y cuando la aplicación modificada esté activa. Si cambiar a otra aplicación, se cambia la configuración para que coincida con esa aplicación. Si vuelve a cambiar a la aplicación modificada restaura su configuración.

Ciertas aplicaciones tienen algunas anulaciones preestablecidas debido, principalmente, a la naturaleza de los cálculos que se realizan en esa aplicación. Por ejemplo, el formato de número en la aplicación Finanzas es Fijo, 2 decimales.

1452 Vista de Configuración de gráfico

ID 1452 · título recuperado de la página española

Muestra la Vista de Configuración de gráfico. Esta vista se utiliza sobre todo para modificar la apariencia de gráficos y el entorno de trazado. No se utiliza en aplicaciones que no incluyen la Vista de gráfico.

1453 Vista Configuración numérica

ID 1453 · título recuperado de la página española

Muestra la vista Configuración numérica. Esta vista se utiliza para determinar la apariencia de la Vista numérica y configurar el factor de zoom.

1454 Comillas simples

ID 1454 · título recuperado de la página española

Oprima la tecla '□' para insertar un par de comillas simples. Las comillas simples se utilizan para introducir expresiones algebraicas.

1455 Llaves

ID 1455 · título recuperado de la página española

{ } : Llaves

Inserta un par de llaves. Las llaves se utilizan para incluir listas.

1456 Corchetes

ID 1456 · título recuperado de la página española

[] : corchetes

Inserta un par de corchetes. Se utilizan para incluir vectores y matrices.

1457 Tecla Eval

ID 1457 · título recuperado de la página española

Sintaxis

EVAL (Expr)

Evalúa la expresión Expr.

1458 Paleta de caracteres especiales

ID 1458 · título recuperado de la página española

Presione Shift 9 (!,∞,→) para mostrar la paleta de caracteres especiales. Esta paleta muestra muchos símbolos matemáticos comunes (como el infinito, promedio y la desviación estándar), así como caracteres griegos.

Presione Esc para cerrar la paleta sin realizar una selección.

Tenga en cuenta que puede mostrar una lista completa de caracteres pulsando Shift + Vars (Chars).

1459 Paleta de relaciones

ID 1459 · título recuperado de la página española

Presione Shift 6 ($\leq, \geq, \neq$) para mostrar la paleta de relaciones. Esta paleta muestra operadores booleanos y relacionales útiles en matemáticas y programación (como $\leq$, $=$, $>$, AND, y OR).

Presione Esc para cerrar la paleta sin realizar una selección.

1460 Menor que

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Menor que.](#)

1461 Menor o igual que

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Menor o igual que.](#)

1462 Mayor que

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Mayor que.](#)

1463 Mayor o igual que

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Mayor o igual que.](#)

1464 Equivalencia

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Equivalencia.](#)

1465 No es igual que

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de No es igual que.](#)

0895 AND lógico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de AND lógico.](#)

1158 NOT lógico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de NOT lógico.](#)

1164 OR lógico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de OR lógico.](#)

1322 XOR lógico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de XOR lógico.](#)

1466 Guión bajo

ID 1466 · título recuperado de la página española

_ : guión bajo

Introduce el carácter de guión bajo (_).

1467 Config. sist. algebraico comp.

ID 1467 · título localizado en HelpEs

Muestra la pantalla de configuración de CAS con varios ajustes para que configure cómo funciona el sistema algebraico computacional.

1468 Tecla Del

ID 1468 · título recuperado de la página española

Presione Shift + Retroceso (Del) para borrar el carácter a la derecha del cursor.

1469 Tecla Help (Ayuda)

ID 1469 · título recuperado de la página española

Muestra la ayuda en línea. La ayuda brindada será relevante a la pantalla que está abierta en el momento que pulsar esta tecla.

La ayuda en línea también proporciona ayuda de todas las teclas en el teclado. También proporciona un árbol de temas de ayuda donde buscar.

Botones de menú:

- [[Árbol]] muestra el árbol de ayuda completo
- [[Teclas]] el sistema de ayuda pasa al modo Tecla; una vez en este modo, pulse cualquier tecla del teclado para obtener ayuda sobre esta
- [-▲ Página ▼-] Página arriba y abajo. Presione el lado derecho para bajar una página y el izquierdo para subirla en un guión de ayuda con varias páginas
- [[OK]] para salir del sistema de ayuda

1470 Teclado del usuario

ID 1470 · título recuperado de la página española

Presione Shift + Help (User) para activar el teclado del usuario (usuario).

Puede asignar funcionalidad alternativa a cualquier tecla del teclado, incluida la funcionalidad proporcionada por las teclas Shift y ALPHA. Esto le permite personalizar el teclado para sus necesidades específicas. Cuando el teclado del usuario está activo, este funciona de la forma que lo haya definido.

Presione User una vez (1U aparece en la barra de título) para activar el teclado del usuario solo la próxima vez que presione la tecla.

Presione User dos veces (↑U aparece en la barra de título) para mantener el teclado del usuario activo. Presione una vez más para desactivarlo.

1471 ≈

ID 1471 · título localizado en HelpEs

$\approx$: Aproximación numérica

Proporciona una aproximación numérica para el elemento seleccionado en el historial de CAS.

Ejemplo

En CAS $1 \div 4 \approx$ devuelve 0,25

0628 Valor de absoluto

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Valor de absoluto.](#)

0886 Coseno inverso

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Coseno inverso.](#)

0897 Última respuesta

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Última respuesta.](#)

0906 Seno inverso

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Seno inverso.](#)

0908 Tangente inversa

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tangente inversa.](#)

0958 Función coseno

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función coseno.](#)

1120 Logaritmo natural

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Logaritmo natural.](#)

1124 Logaritmo general

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Logaritmo general.](#)

1155 Número negativo

ID 1155 · título recuperado de la página española

- Valor
- Expresión

Menos unario.

Cambia el signo del Valor o de la Expresión. Se usa para introducir números negativos.

1159 Tecla de raíz n-ésima

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tecla de raíz n-ésima.](#)

1274 Seno

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Seno.](#)

1406 Plantilla de matemáticas

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Plantilla de matemáticas.](#)

1539 Caracteres matemáticos

ID 1539 · título localizado en HelpEs

Presione Shift 9 (!,∞,→) para abrir un menú de símbolos matemáticos y caracteres griegos usados comúnmente. Todos estos caracteres también se pueden encontrar al presionar Shift Vars (Chars).

1472 Aritmética de enteros

ID 1472 · título localizado en HelpEs

Puede realizar aritmética de enteros en cuatro bases: decimal (base 10), binaria (base 2), octal (base 8) y hexadecimal (base 16). Por ejemplo, podría multiplicar 4 en base 16 por 71 en la base 8 y la respuesta es E4 en base 16. Esto es equivalente en base 10 a multiplicar 4 por 57 para obtener 228.

Indica que está a punto de realizar aritmética de enteros al insertar el símbolo numeral: # (presione ALPHA 3) antes del número. A continuación, indica qué base utilizar para el número agregando el marcador de base adecuado.

d: decimal

b: binario

o: octal

h: hexadecimal

[blanco]: base predeterminada

1473 Cambiar la base de números enteros

ID 1473 · título localizado en HelpEs

La base predeterminada de la calculadora para aritmética con números enteros es 16 (hexadecimal).

Para cambiar la base predeterminada:

1. Presione Shift Inicio (Configuración de inicio).
2. Seleccione la base que desea en el menú Enteros: Binario, Octal, Decimal o Hex.
3. El campo a la derecha del menú de Enteros es el campo de tamaño de palabra. Esta es la cantidad máxima de bits que pueden representar un entero. El valor predeterminado es 32, pero puede cambiarlo por cualquier valor entre 1 y 64.
4. Si quiere permitir enteros con signo, seleccione la opción $\pm$ a la derecha del campo tamaño de palabra. Si elige esta opción se reduce el tamaño máximo de un entero a un bit menos que el tamaño de palabra.

1474 Manipular números enteros

ID 1474 · título localizado en HelpEs

La tecla Base abre la pantalla Editar entero. Si el elemento seleccionado actualmente (o el resultado más reciente) es un número entero, ese valor se carga en la pantalla para su edición.

En el cuadro de diálogo Editar entero, el campo Era en la parte superior muestra el entero original que seleccionó en la Vista de inicio. El campo Salida muestra el número entero editado. Ambos números enteros se muestran inicialmente en la base predeterminada, tal como se especifica en la página 1 de la Configuración de inicio.

El campo 16 es la representación hexadecimal de la Salida.

El campo 10 es la representación decimal de la Salida.

El cuadro debajo del valor decimal muestra la representación binaria (bit) de 64 bits de la Salida.

Cambiar el valor por cualquiera de los siguientes métodos actualiza el valor de la Salida y todas las representaciones:

- Desplace el número entero a la izquierda o a la derecha. Se pierden los bits desplazados en ambos extremos. Arrastre la pantalla hacia la izquierda o hacia la derecha para desplazarse 1 bit en la dirección correspondiente. Mueva la rueda a la izquierda/derecha para cambiar 1 bit. Mueva la rueda alfa a la derecha/izquierda para cambiar 4 bits (1 nibble [cantidad de datos de 4 bits]). Desplace la rueda a la derecha/izquierda 8 bits (un byte).
- Cambie el tamaño de la palabra (número de bits) del número entero. Arrastre la pantalla hacia abajo/arriba para aumentar/disminuir 1. Presione la rueda hacia arriba/abajo para aumentar/disminuir 1. Mueva la rueda alfa hacia arriba/abajo para aumentar/disminuir 4 bits (1 nibble [cantidad de datos de 4 bits]). Desplace la rueda hacia arriba/abajo para aumentar/disminuir 8 bits (1 byte).
- Cambie el signo del entero: Presione la tecla +/-
- Desplácese por los ajustes de base (sistema, hexadecimal, decimal, octal y binario): teclas + y -

Botones de menú:

- [[Rein.]] devuelve todos los cambios a su estado original

- **[[Base]]** pasa por las distintas bases; como ocurre al presionar +
 - **[[Firmada]]** cambia el tamaño de las palabras entre con signo y sin signo
 - **[[NO]]** devuelve el complemento de uno, (es decir, cada bit del tamaño de palabra especificado se invierte: 0 se sustituye por 1 y 1 por 0)
 - **[[Editar]]** activa el modo de edición. El modo de edición está activo si aparece un carácter de viñeta en el botón **[[Editar]]**. Cuando el modo de edición está activo, un selector de dígitos resalta un solo dígito y puede desplazarse por el cuadro de diálogo con ayuda de la rueda. Los campos hex y decimal se pueden modificar, al igual que la representación de bits, un dígito por vez. Un cambio en un campo modifica automáticamente el resto de campos.
 - **[[OK]]** cierra el cuadro de diálogo y guarda sus cambios. Si no desea guardar los cambios, presione Esc.
- Valide el cambio en el número mediante **[[OK]]** o bien, cancele el cambio mediante la tecla Esc.

1475 Catálogo de lista

ID 1475 · título localizado en HelpEs

Presione Shift + 7 (List) para ver el catálogo de la lista. Hay diez listas disponibles, llamadas L1-L9 y L0.

Utilice la rueda hacia arriba/abajo para seleccionar un nombre de lista.

Pulse [[Editar]] o presione Enter para editar la lista seleccionada.

Botones de menú:

- [[Editar]] abre la lista seleccionada para su edición en el Editor de lista
- [[Elimi.]] permite eliminar el contenido de la lista resaltada
- [[Enviar]] cuando está presente, envía una lista a otra HP Prime

1476 Editor de lista

ID 1476 · título localizado en HelpEs

El Editor de listas se ha diseñado para ayudarlo a crear y editar cualquiera de las listas disponibles en la calculadora HP Prime. Estas listas predeterminadas se denominan L1-L9 y L0.

Al seleccionar una lista, se abre el Editor de listas. Aquí es donde puede añadir o cambiar elementos de una lista. Al abrir una lista por primera vez, esta estará en blanco. Para introducir un elemento, simplemente comience a introducirlo. Los elementos de menú en el Editor de listas son:

- [[Editar]] copia el elemento seleccionado en la línea de entrada donde se puede editar. Este elemento solo está visible cuando se selecciona un elemento en la lista.
- [[Más]] abre un menú con opciones para editar la lista
- [[Ir a]] pasa a un elemento específico en la lista. Útil para listas muy grandes.
- [[Ir]] alterna cómo se mueve el cursor cuando se presiona Enter. Las opciones son Abajo, Derecha, y Ninguno.

El menú [[Más]] del Editor de listas contiene las siguientes opciones para editar una lista:

- Insertar Fila: inserta una nueva fila en la lista actual. La nueva fila contiene 0 como su elemento.
- Eliminar Columna: borra el contenido de la lista actual. Para eliminar un solo elemento, selecciónelo y pulse la tecla Retroceso.
- Seleccionar Fila: selecciona la fila actual. Una vez seleccionada, la fila se puede copiar. Columna: selecciona la lista actual. Una vez seleccionada, la lista se puede copiar. Cuadro: abre un cuadro de diálogo para seleccionar una matriz rectangular definida por un punto de partida y una ubicación final. También puede mantener pulsada una celda para iniciar la selección, y a continuación, arrastrar para seleccionar una matriz rectangular de elementos. Una vez seleccionada, la matriz se puede copiar. Intercambiar final: intercambia la celdas de inicio y final de la matriz de celdas rectangular seleccionada.
- Selección: activa y desactiva el modo de selección. También puede mantener pulsada una celda, y a continuación, arrastrar para seleccionar.
- Interc Columna: intercambia el contenido de dos columnas (listas).

Escriba en la primera entrada en la lista y pulse la tecla Enter. Continúe hasta que haya completado la lista. Cuando haya completado la lista, podrá volver al Catálogo de listas.

En programas o en la Vista de inicio, puede hacer referencia a su lista por nombre (L1, L2, etc.) para realizar operaciones en su nueva lista. Utilice la rueda hacia la izquierda/derecha para desplazarse por las diez listas una vez que esté en el Editor de listas.

1477 Catálogo de matriz

ID 1477 · título localizado en HelpEs

Pulse Shift 4 (Matrix) para acceder al catálogo de matrices.

Hay diez matrices disponibles, denominadas M1-M9 y M0.

Botones de menú:

- **[[Editar]]** abre la matriz resaltada para su edición en el editor de matrices
- **[[Elimi.]]** elimina el contenido de una matriz
- **[[Vector]]** convierte una matriz en un vector
- **[[Enviar]]** cuando está presente, envía una matriz a otra HP Prime

1478 Editor de matriz

ID 1478 · título localizado en HelpEs

Cuando se selecciona una matriz, se abre el Editor de matriz. Los elementos de menú en el Editor de matriz son:

- **[[Editar]]** copia el elemento seleccionado en la línea de entrada donde se puede editar. Este elemento solo está visible cuando se selecciona un elemento en la matriz o el vector.
- **[[Más]]** abre un menú con opciones para la edición de la matriz
- **[[Ir a]]** pasa a un elemento específico en la matriz. Útil para grandes matrices.
- **[[Ir]]** alterna cómo se mueve el cursor cuando se presiona Enter. Las opciones son Abajo, Derecha, y Ninguno.

El menú **[[Más]]** del Editor de matriz contiene opciones similares a las del menú **[[Más]]** del Editor de listas, pero con más opciones apropiadas para la edición de matrices. Las opciones son las siguientes:

- **Insertar Fila:** inserta una nueva fila en la matriz actual, por encima de la fila actual. La nueva fila contiene ceros.
- **Columna:** inserta una nueva columna en la matriz, a la izquierda de la columna actual. La nueva columna contiene ceros.
- **Eliminar Fila:** elimina la fila actual de la matriz. **Columna:** elimina la columna actual de la matriz. **Todo:** elimina el contenido de la matriz.
- **Seleccionar Fila:** selecciona la fila actual. Una vez seleccionada, la fila se puede copiar. **Columna:** selecciona la columna actual. Una vez seleccionada, la columna se puede copiar. **Cuadro:** abre un cuadro de diálogo para seleccionar una matriz rectangular definida por un punto de partida y una ubicación final. También puede mantener pulsada una celda para iniciar la selección, y a continuación, arrastrar para seleccionar una matriz rectangular de elementos. Una vez seleccionada, la matriz se puede copiar. **Intercambiar final:** intercambia las celdas de inicio y fin de la matriz de celdas seleccionada.
- **Selección:** activa y desactiva el modo de selección. También puede mantener pulsada una celda, y a continuación, arrastrar para seleccionar.
- **Interc Fila:** intercambia el contenido de dos filas. **Columna:** intercambia el contenido de dos columnas.

Usted no tiene que definir las dimensiones de una matriz primero; basta empezar a escribir los valores. Puede introducir los valores fila por fila, o columna por columna; el botón **[[Ir]]** alterna entre las opciones.

Como ocurre con las listas, puede enviar la matriz a otra calculadora HP Prime o recibir una matriz de otra calculadora HP Prime. En los programas o en la vista de Inicio, puede hacer referencia a estas matrices por nombre para hacer operaciones con ellas.

1479 Catálogo de programas

ID 1479 · título localizado en HelpEs

Presione Shift 1 (Program) para acceder al catálogo de programas.

En el Catálogo de programas, puede ver la lista de programas existentes, editar o ejecutar cualquiera de ellos, o enviarlos a otra calculadora HP Prime. También sirve para crear un nuevo programa. Un programa HP Prime puede ser tan simple como una sola función definida por el usuario, o puede contener un conjunto de funciones relacionadas que se exportan para aparecer como un submenú en el menú de usuario de los menús de cuadro de herramientas. Las variables de estas funciones se pueden exportar para que aparezcan en el menú de usuario en la tecla Vars o de forma local. Un programa HP Prime también podría ser una aplicación completa por sí misma. La elección es suya.

Botones de menú:

- **[[Editar]]** abre la lista resaltada para su edición en el editor de programas
- **[[Nuevo]]** solicita un nuevo nombre de programa y, a continuación, abre el editor de programas
- **[[Más]]** muestra más opciones de menú (Guard., Renom., Orden., Elim. y Borrar) para el programa seleccionado
- **[[Enviar]]** envía un programa a otra calculadora HP Prime que admita conectividad de unidad a unidad
- **[[Depur.]]** depura el programa resaltado
- **[[Ejecu.]]** ejecuta el programa resaltado

Pulse **[[Nuevo]]** para crear un nuevo programa e introducir un nombre para este. Utilice la rueda hacia arriba/abajo para seleccionar un programa existente. Pulse **[[Editar]]** para abrir el Editor de programas y editar un programa existente.

Un nombre de programa solo puede contener caracteres alfanuméricos (letras y números) y el carácter de subrayado. El primer carácter debe ser una letra. Por ejemplo, GOOD_NAME, RollDice y Spin2 son nombres de programa válidos.

1480 Editor de programa

ID 1480 · título localizado en HelpEs

Una vez introduzca el nombre del programa y toque **[[OK]]**, se abrirá el editor de programas. Aquí, se creará una plantilla para el programa. La plantilla consta de un encabezado para una función del mismo nombre que el programa y un par BEGIN-END que encierra las instrucciones de la función.

Botones de menú:

- **[[Cmds]]** abre un menú desde el que puede seleccionar comandos de programación comunes
- **[[Plant]]** abre un menú desde el que puede seleccionar distintas plantillas para estructuras de controles de programación comunes
- **[-▲ Página ▼-]** le permite desplazarse de una página a otra en un programa de varias páginas
- **[[Compr]]** selecciona el programa para comprobar errores de sintaxis

Si pulsa la tecla de Menú en el Editor de programas, verá dos nuevas opciones:

- **Crear tecla de usuario:** toque esta opción y pulse cualquier tecla para pegar una plantilla en su programa para redefinir esa tecla como una tecla de usuario.
- **Insertar pragma.:** toque esta opción para pegar una definición de modo `#pragma` en su programa. La definición del modo `#pragma` es la siguiente: `#pragma mode(separator(), integer())` Utilice la definición de modo `#pragma` para definir el conjunto de separadores utilizados para agrupar dígitos y el tipo de entero. La definición del modo `#pragma` obligará al programa a compilar con esta configuración. Esta función es útil para adaptar un programa escrito para una cultura que utiliza diferentes símbolos de agrupación (. vs.,) que la suya.

Puede escribir en su programa letra a letra si conoce los nombres de los comandos o utilizar el menú **[[Cmds]]** o **[[Plant]]**. También puede pulsar la tecla del cuadro de herramientas para visualizar más funciones y comandos. Una vez finalizado el programa, pulse Shift 1 (Prgm) para volver al catálogo de programas. Desde el catálogo, puede enviar el programa a otra calculadora HP Prime o recibir un programa de otra calculadora HP Prime. Introduzca el nombre del programa en la vista de Inicio y pulse Enter para ejecutar el programa.

Nota: cuando sale del editor, este guarda los cambios automáticamente. Si desea guardar la versión original del programa antes de realizar cambios, asegúrese de utilizar el botón **[[Más]]** en el Catálogo de programas y seleccionar Guardar.

1481 Menú comandos

ID 1481 · título localizado en HelpEs

El menú `[[Cmds]]` contiene los comandos de programación de la calculadora HP Prime. Este menú principal contiene categorías de comandos y el submenú contiene los comandos específicos de cada categoría. Utilice la rueda para seleccionar una categoría del menú principal y, a continuación, utilice la rueda hacia la derecha para desplazarse por el submenú de comandos de dicha categoría. Utilice la rueda hacia la derecha/izquierda para seleccionar el comando deseado de la categoría.

Un programa de HP Prime contiene una secuencia de comandos que se ejecutan automáticamente para realizar una tarea.

Los comandos se separan con punto y coma (;). Los comandos que aceptan varios argumentos incluyen a estos últimos entre paréntesis, separados por comas (,). Por ejemplo,

`PIXON(xposition, yposition);`

En algunos casos, los argumentos de comandos son opcionales. Si se omite un argumento opcional, se utiliza un valor predeterminado en su lugar. Los argumentos de comandos opcionales se muestran entre [corchetes]. Los corchetes también se utilizan con vectores y normalmente no son opcionales.

Los programas pueden contener cualquier número de subrutinas (cada una de las cuales es una función o procedimiento). Las subrutinas empiezan con un encabezamiento, que consta del nombre, seguido de paréntesis y una lista de parámetros o argumentos, separados por comas. El cuerpo de una subrutina consta de una secuencia de instrucciones incluidas entre un par `BEGIN END`. Una función puede devolver un valor por medio del comando `RETURN`.

1482 Cadena

ID 1482 · título localizado en HelpEs

Este menú contiene todos los comandos de manipulación de cadenas. Una cadena es un conjunto de caracteres encerrados en comillas dobles; por ejemplo, `"ABC"`, `"12A"` y `"3-A"` son todas cadenas.

El carácter `\` empieza una secuencia de escape y el(los) carácter(es) inmediatamente posteriores se interpretan de forma especial, según se describe a continuación.

- `""`: `"`
- `\"`: `"`
- `\\`: `\`
- `\0`: 0
- `\n`: nueva línea
- `\r`: retorno de carro
- `\t`: tabulación
- `\nnn`: octal, como en `\115` para el carácter 77
- `\xnn`: hex, como en `\x45` para el carácter 69
- `\unnnn`: Unicode para el carácter `nnnn`

Para poner una nueva línea en una cadena, también puede presionar Enter para cerrar el texto en ese punto.

1483 Reemplazar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Reemplazar.](#)

0905 ASC — Devuelve una lista que contiene los valores Unicode numéricos de la cadena.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de ASC.](#)

0936 CHAR — Devuelve la cadena correspondiente a los códigos de carácter Unicode numéricos en Lista o Vector, o el...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de CHAR.](#)

0973 Dimensiones de la cadena

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dimensiones de la cadena.](#)

1002 Evaluar cadena

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Evaluar cadena.](#)

1077 En cadena

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de En cadena.](#)

1108 Parte izquierda

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Parte izquierda.](#)

1129 Minúscula

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Minúscula.](#)

1140 Medio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Medio.](#)

1282 Cadena del identificador

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Cadena del identificador.](#)

1283 Rotar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Rotar.](#)

1284 Parte derecha

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Parte derecha.](#)

1285 STRING — Evalúa Expresión y devuelve el resultado como una cadena.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de STRING.](#)

1739 UPART — Devuelve la parte de unidad de un objeto de unidad.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de UPART.](#)

1312 Mayúsculas

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Mayúsculas.](#)

1484 Dibujo

ID 1484 · título localizado en HelpEs

Este menú contiene comandos para dibujar primitivas gráficas simples, como líneas, rectángulos y arcos, en variables gráficas. La HP Prime dispone de 10 variables gráficas, identificadas como G0 a G9. Estas variables también se denominan objetos gráficos, o GROBs para abreviar. G0 representa siempre la pantalla gráfica actual y se utiliza como GROB predeterminado cuando no se especifica ningún GROB de destino en un comando de dibujo.

Las variables G1 a G9 pueden almacenar temporalmente objetos gráficos para su uso en aplicaciones y se borran cuando se restablece la calculadora mediante la combinación de teclas On+Symb o el orificio de reinicio situado en la parte posterior. Tenga en cuenta que estas variables son globales y se comparten entre aplicaciones.

Hay dos conjuntos idénticos de funciones para modificar variables gráficas. El primer conjunto utiliza coordenadas cartesianas basadas en el plano de coordenadas definido en la aplicación actual por las variables Xmin, Xmax, Ymin e Ymax en Configuración de Gráfico. El segundo conjunto utiliza coordenadas absolutas de píxeles basadas en la pantalla de 320×240 de la HP Prime. Para estas funciones, (0,0) es el píxel superior izquierdo del GROB y (319,239) es el píxel inferior derecho. Las funciones basadas en píxeles tienen el sufijo _P, como ARC_P y LINE_P.

Muchos comandos de dibujo permiten especificar explícitamente el color como un valor entero. Para los enteros binarios, una representación hexadecimal práctica es #AARRGGBBh, donde AA, RR, GG y BB son, respectivamente, los componentes alfa, rojo, verde y azul de 8 bits.

El componente alfa controla la transparencia: 0 (#00h) representa un píxel completamente opaco, mientras que 255 (#FFh) representa un píxel completamente transparente. Los valores alfa intermedios producen colores semitransparentes, lo que permite que los objetos dibujados se mezclen con el fondo y revelen parcialmente el contenido situado debajo.

Un color también puede expresarse como una lista: {color, alpha}, donde color es un valor RGB de 24 bits y alpha es un valor de 0 a 255, siendo 0 completamente opaco y 255 completamente transparente. La función RGB, RGB(R,G,B[,A]), puede utilizarse para construir un valor de color a partir de sus componentes rojo, verde, azul y, opcionalmente, alfa.

Si no se especifica ningún color para un comando de dibujo, se utiliza negro o blanco de forma predeterminada, dependiendo de la opción seleccionada en Tema del color, en Configuración de Inicio, página 2.

Sin embargo, tenga en cuenta que, en algunos comandos, la transparencia se especifica como un parámetro separado en lugar de formar parte del valor de color. En tales casos, consulte la documentación del comando correspondiente para determinar cómo se aplica la transparencia.

Los GROBs G1 a G9 pueden definirse con cualquier dimensión e inicializarse con un color o con datos gráficos mediante los comandos DIMGROB o DIMGROB_P. Ambos comandos permiten especificar el nivel de transparencia de los píxeles inicializados.

Cuando se copia un GROB, o una parte de un GROB, en otro GROB, por ejemplo mediante los comandos BLIT o BLIT_P, sus píxeles afectan al destino de acuerdo con su transparencia. Los píxeles completamente transparentes no sobrescriben los píxeles correspondientes del destino, mientras que los píxeles parcialmente transparentes se mezclan con ellos.

En las descripciones de sintaxis de los comandos individuales, los puntos suelen mostrarse como pares de coordenadas x, y. En muchos comandos, a menos que se indique lo contrario, un punto también puede pasarse como un único argumento en una de las siguientes formas: una lista de dos elementos {x,y}, un número complejo en forma rectangular $x+yi$, la forma de punto (x,y) o la forma polar $r\angle\theta$.

Por ejemplo, las siguientes formas representan el mismo punto:

- 10, 20
- {10, 20}
- $10+20i$

Cuando un comando acepta una lista de puntos, estas formas pueden combinarse libremente dentro de la misma lista.

1485 PX→C — Transforma las coodernadas de píxel en coordenadas cartesianas.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de PX→C.](#)

1486 RGB — Devuelve un valor de color que se puede usar como parámetro de color de las funciones de dibujo.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de RGB.](#)

1487 Píxeles

ID 1487 · título localizado en HelpEs

En esta sección se enumeran los comandos para dibujar utilizando coordenadas de píxel.

0902 Dibujar arco

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dibujar arco.](#)

0923 Copiar GROB

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Copiar GROB.](#)

0975 Dimensiones de GROB

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dimensiones de GROB.](#)

1100 Dibujar polígono relleno

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dibujar polígono relleno.](#)

1043 Obtener píxel de color

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Obtener píxel de color.](#)

1047 Altura GROB

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Altura GROB.](#)

1049 Ancho de GROB

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ancho de GROB.](#)

1081 GROB Invertida

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de GROB Invertida.](#)

1114 Dibujo de líneas

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dibujo de líneas.](#)

1275 Píxel Off

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Píxel Off.](#)

1276 Píxel On

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Píxel On.](#)

1277 Rectángulo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Rectángulo.](#)

1283 Rotar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Rotar.](#)

1278 Copiar GROB en el Destino

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Copiar GROB en el Destino.](#)

1280 Medir el tamaño del texto

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Medir el tamaño del texto.](#)

1279 Dibujar texto

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dibujar texto.](#)

1281 Dibujar triángulo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dibujar triángulo.](#)

1488 Cartesiano

ID 1488 · título localizado en HelpEs

Los comandos para dibujar usando coordenadas cartesianas se enumeran en esta sección.

0901 Dibujar arco

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dibujar arco.](#)

0922 Copiar GROB

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Copiar GROB.](#)

0974 Dimensiones de GROB

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dimensiones de GROB.](#)

1101 Dibujar polígono relleno

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dibujar polígono relleno.](#)

1042 Obtener píxel de color

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Obtener píxel de color.](#)

1046 Altura GROB

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Altura GROB.](#)

1048 Ancho de GROB

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ancho de GROB.](#)

1080 GROB Invertida

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de GROB Invertida.](#)

1113 Dibujo de líneas

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dibujo de líneas.](#)

1286 Píxel Off

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Píxel Off.](#)

1287 Píxel On

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Píxel On.](#)

1288 Dibujar rectángulos

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dibujar rectángulos.](#)

1289 Copiar GROB en el Destino

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Copiar GROB en el Destino.](#)

1290 Dibujar texto

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dibujar texto.](#)

1489 Dibujar triángulo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dibujar triángulo.](#)

1490 ICONO del teclado

ID 1490 · título recuperado de la página española

Sintaxis

ICON name hexPngData

Un ICON es un objeto gráfico con nombre creado a partir de datos de imagen PNG codificados en hexPngData. Se define en el nivel superior de un programa, fuera de cualquier función.

Una vez definido, un ICON puede utilizarse en funciones gráficas haciendo referencia a su nombre, por ejemplo:

```
BLIT_P(G0, "name")
```

Los ICON están pensados principalmente para utilizarse como objetos gráficos de origen, aunque su contenido puede modificarse.

El tamaño de un ICON no puede modificarse. Por lo tanto, un ICON no puede utilizarse como destino de SUBGROB(_P) ni de DIMGROB(_P).

Los ICON se vuelven a crear cada vez que un programa se carga de nuevo desde el almacenamiento o se recompila. Como resultado, los cambios realizados en ellos no son permanentes, al igual que los cambios realizados en las variables EXPORT.

0966 C→PX — Convierte las coordenadas cartesianas en coordenadas de pantalla.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de C→PX.](#)

0978 Dibujar menú de botón

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Dibujar menú de botón.](#)

1031 Congelación de pantalla

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Congelación de pantalla.](#)

1491 Matriz

ID 1491 · título localizado en HelpEs

Los comandos de Matriz permiten que se manipulen las matrices desde dentro de un programa. En esta sección de ayuda, matrixname se refiere al nombre de una matriz y debe ser M0, M1, M2, ..., M9.

0887 Agregar columna

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Agregar columna.](#)

0891 Agregar fila

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Agregar fila.](#)

0968 Eliminar columna

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Eliminar columna.](#)

0970 Eliminar fila

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Eliminar fila.](#)

1014 Editar matriz

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Editar matriz.](#)

1291 Redimensionar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Redimensionar.](#)

1292 SCALEADD — Multiplica la fila1 especificada del nombre de la matriz por valor.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de SCALEADD.](#)

1293 Extraer parte

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Extraer parte.](#)

1294 Intercambiar columnas

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Intercambiar columnas.](#)

1295 Intercambiar filas

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Intercambiar filas.](#)

1296 SCALE — Multiplica una fila de una matriz por un escalar.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de SCALE.](#)

1483 Reemplazar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Reemplazar.](#)

1492 Funciones de aplicación

ID 1492 · título localizado en HelpEs

Este menú contiene comandos para configurar aplicaciones dentro de un programa.

1493 Marcar (seleccionar) definición

ID 1493 · título recuperado de la página española

Sintaxis

CHECK(n)

Marca (selecciona) el campo de la definición simbólica correspondiente en la aplicación actual. El entero n debe estar entre 0 y 9 para la mayoría de aplicaciones. Para las aplicaciones de Estadísticas 1-Var y Estadísticas 2-Var, n debe estar entre 1 y 5.

Por ejemplo, CHECK(3) activa F3 si la aplicación actual es Función. Entonces aparecería una marca de verificación junto a F3 en la Vista simbólica, F3 se traza en la Vista de gráfico y se evalúa en la Vista numérica.

1494 Is Checked (seleccionado)

ID 1494 · título recuperado de la página española

Sintaxis

ISCHECK(n)

Devuelve 1 o 0 dependiendo de si el campo de la definición simbólica correspondiente en la aplicación actual está seleccionada o no. El entero n debe estar entre 0 y 9 para la mayoría de las aplicaciones. Para las aplicaciones 1-Var estadística y 2-Var estadística, n debe estar entre 1 y 5.

1495 Inicia la aplicación

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Inicia la aplicación.](#)

1496 Desactivar (anular selección) definición

ID 1496 · título recuperado de la página española

Sintaxis

UNCHECK(n)

Desactiva (anula la selección de) la definición correspondiente en la Vista simbólica. Para la mayoría de las aplicaciones el entero n debe ser un valor entre 0 y 9. Para las aplicaciones 1-Var estadística y 2-Var estadística, n debe estar entre 1 y 5.

Por ejemplo, UNCHECK(3) desmarca F3 si la aplicación actual es Función.

1497 Palabra clave View

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Palabra clave View.](#)

1498 Entero

ID 1498 · título localizado en HelpEs

Este menú contiene comandos que trabajan con números enteros.

0916 AND de bit lógico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de AND de bit lógico.](#)

0917 NOT de bit lógico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de NOT de bit lógico.](#)

0918 OR de bit lógico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de OR de bit lógico.](#)

0919 Desplazamiento lógico de bits hacia la izquierda

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Desplazamiento lógico de bits hacia la izquierda.](#)

0920 Desplazamiento lógico de bits hacia la derecha

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Desplazamiento lógico de bits hacia la derecha.](#)

0921 XOR de bit lógico

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de XOR de bit lógico.](#)

0927 Base a real

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Base a real.](#)

1733 GET — Recupera un objeto de un objeto composite (lista, matriz, cadena o entero binario) utilizando el índice...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de GET.](#)

1039 Obtener Base

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Obtener Base.](#)

1040 Obtener bits

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Obtener bits.](#)

1297 Real a base

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Real a base.](#)

1298 Configurar Base

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Configurar Base.](#)

1299 Configurar Bits

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Configurar Bits.](#)

1499 E/S

ID 1499 · título localizado en HelpEs

Los comandos de entrada/salida permiten que los usuarios interactúen con los programas.

0938 Cuadro de elección

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Cuadro de elección.](#)

0980 Editar lista

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Editar lista.](#)

0981 Editar matriz

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Editar matriz.](#)

1041 Obtener tecla

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Obtener tecla.](#)

1075 Formulario de entrada

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Formulario de entrada.](#)

1089 Se presiona la tecla

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Se presiona la tecla.](#)

1145 Evento Get Touch

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Evento Get Touch.](#)

1147 Cuadro de mensaje

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Cuadro de mensaje.](#)

1300 Iniciara Vista

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Iniciara Vista.](#)

1301 Print

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Print.](#)

1740 PRINT2D — Imprime el resultado de expr o una cadena en la terminal.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de PRINT2D.](#)

1328 Esperar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Esperar.](#)

1727 USBOpen — () cierra cualquier conexión activa y devuelve la lista de los dispositivos HID actualmente conectados.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de USBOpen.](#)

1728 USBSend — Envía un paquete USB compuesto por los números proporcionados en la lista de datos (un elemento es un...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de USBSend.](#)

1729 USBRecv — Devuelve una lista de números que representa los datos que provienen del dispositivo USB.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de USBRecv.](#)

1500 Más

ID 1500 · título localizado en HelpEs

Este menú contiene comandos de programación adicionales.

0929 Evaluación CAS

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Evaluación CAS.](#)

0994 Ejecutar sobre el elemento

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Ejecutar sobre el elemento.](#)

1015 Evaluar la lista

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Evaluar la lista.](#)

1059 →HMS — Muestra un valor decimal en formato sexagesimal.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de →HMS.](#)

1060 HMS→ — Muestra un valor sexagesimal en formato decimal.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de HMS→.](#)

1090 Iterar expresión

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Iterar expresión.](#)

1741 Terminal — La función Terminal te da acceso a los datos del terminal y te permite controlarlos...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Terminal.](#)

1302 Evaluación del tiempo

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Evaluación del tiempo.](#)

1303 Valor de ticks internos

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Valor de ticks internos.](#)

1304 Tipo de objeto

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Tipo de objeto.](#)

1343 Porcentaje de cambio

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Porcentaje de cambio.](#)

1344 Porcentaje Total

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Porcentaje Total.](#)

1730 Resource — La función Resource permite acceder a los recursos binarios almacenados como parte del programa desde el...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Resource.](#)

1501 Plantilla

ID 1501 · título localizado en HelpEs

Este menú contiene toda la estructura y las plantillas de programación que son útiles cuando se edita un programa

1502 Bloque

ID 1502 · título localizado en HelpEs

Este menú contiene comandos para ingresar estructuras de bloque en los programas.

0910 Bloque BEGIN END

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Bloque BEGIN END.](#)

1102 Detener ejecución

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Detener ejecución.](#)

1305 Comando Return

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Comando Return.](#)

1503 Bifurcación

ID 1503 · título localizado en HelpEs

Este menú contiene comandos de bifurcación comunes como IF ... THEN.

0930 CASE — Inicia una estructura de rama "CASE ... END".

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de CASE.](#)

1069 Estructura de bifurcación IF

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Estructura de bifurcación IF.](#)

1070 Estructura de la captura de error

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Estructura de la captura de error.](#)

1504 Variable

ID 1504 · título localizado en HelpEs

Este menú contiene las opciones relativas a las variables que se pueden crear en los programas.

1001 Función o variables EXPORT

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Función o variables EXPORT.](#)

1732 CONST — En un programa, la palabra clave CONST puede utilizarse para declarar constantes globales a nivel de...

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de CONST.](#)

1016 Asignar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Asignar.](#)

1123 Palabra clave LOCAL

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Palabra clave LOCAL.](#)

1306 Almacenar

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Almacenar.](#)

1505 Bucle

ID 1505 · título localizado en HelpEs

Este menú contiene comandos de bucle como FOR ... NEXT.

1506 Repetir estructura de bucle

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Repetir estructura de bucle.](#)

0925 Romper bucle

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Romper bucle.](#)

0952 CONTINUE — Transfiere la ejecución en un bucle al principio de la siguiente iteración del enésimo bucle superior.

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de CONTINUE.](#)

1027 Estructura de bucle For

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Estructura de bucle For.](#)

1329 Estructura de bucle while

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Estructura de bucle while.](#)

1508 Palabra clave de tecla

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Palabra clave de tecla.](#)

1497 Palabra clave View

Referencia cruzada

Esta entrada también aparece en otra rama del árbol. [Ir a la ficha principal de Palabra clave View.](#)

1509 Entorno de depuración

ID 1509 · título localizado en HelpEs

Una vez que toque la tecla de menú `[[Depur.]]` con un programa seleccionado, si un programa alcanza una instrucción `DEBUG()` o si depura un programa usando el comando `DEBUG()`, se inicia el entorno de depuración.

Este entorno consta de tres partes:

- Una barra de título en la parte superior, con el nombre actual del programa y/o de la rutina que se está depurando
- El listado del programa que se está depurando, si está disponible; de lo contrario, muestra la instrucción actual y la siguiente
- Una lista de observación de variables en una tabla de dos columnas

En la lista de variables, escriba en la columna izquierda los nombres de las variables que desea observar durante la depuración. Sus valores actuales se mostrarán en la columna derecha. Al escribir un nuevo valor en la columna derecha, puede cambiar el valor de una variable.

Las teclas de menú son:

- `[[Editar]]` edita el elemento o valor seleccionado.
- `[[Omitir]]` ejecuta una subrutina, pero no la depura.

- `[[Incrm.]]` avanza al siguiente paso del programa; si el siguiente paso es una subrutina, entra en ella y comienza a depurarla.
- `[[Intercam.]]` cambia a la pantalla de la calculadora para que pueda ver la salida del programa. Pulse cualquier tecla para volver al depurador.
- `[[Final]]` sale del entorno de depuración y vuelve al Catálogo de programas.
- `[[Cont]]` continúa la ejecución del programa sin depuración.

Si la lista de variables tiene el foco, `[[Editar]]` puede usar para cambiar la variable indicada en una fila o el valor de la variable.

Si el listado del programa tiene el control, `[[Editar]]` detiene la evaluación del programa y pasa al editor de programas para permitirle modificar el programa.

Puede arrastrar el código fuente del programa.

Puede arrastrar el encabezado de la lista de variables para ver más o menos del programa según sea necesario.

1533 Más

ID 1533 · título localizado en HelpEs

El menú Mas del Editor de programas brinda acceso a una funcionalidad menor asociada con los programas y con la edición de un programa.

1731 Recursos del programa

ID 1731 · título localizado en HelpEs

Un programa puede contener más que su código fuente principal. Se pueden almacenar datos y código fuente adicionales junto con el programa en forma de recursos. Los recursos forman parte del programa y pueden crearse, editarse, copiarse o eliminarse desde el menú `[[Más]]` del Editor de programas.

Los recursos pueden contener código fuente adicional o datos binarios.

Recursos de código fuente

Los recursos de código fuente pueden utilizarse para dividir un programa grande en secciones más pequeñas y manejables. Pueden contener código fuente HP PPL, CAS o Python.

Recursos binarios

Los recursos binarios pueden utilizarse para almacenar datos binarios arbitrarios o imágenes como parte de un programa. Los tipos disponibles de recursos binarios son Binary asset, Png image y Jpg image.

Al crear o editar un recurso de tipo Png image o Jpg image, el contenido actual de G1 puede copiarse al recurso desde el menú `[[Más]]`. La imagen se almacena en formato PNG o JPEG; el recurso en sí no es un GROB.

Los recursos binarios también pueden leerse y modificarse desde un programa mediante la función Resource. Consulte Resource para obtener más información.

Administración de recursos

Utilice el menú `[[Más]]` del Editor de programas para crear y administrar recursos.

Toque Archivos > Nuevo programa para añadir un recurso al programa. Introduzca un nombre y seleccione su tipo.

Toque Copiar para crear una copia del recurso actual con un nuevo nombre.

Toque Elim. para eliminar el recurso actual.

Todos los recursos pertenecientes al programa aparecen en el menú `[[Más]]`. El recurso que se está editando actualmente aparece marcado. Seleccione otro recurso para editarlo.

1534 Marcador

ID 1534 · título localizado en HelpEs

Los marcadores son ubicaciones del cursor que se guardan en un código fuente del programa. Usted puede moverse rápidamente entre ellos para acelerar la edición del programa.

Para establecer un marcador, pulse un número de marcador no utilizado. Vuelva a pulsar este marcador para mover el cursor a esa ubicación de marcador. Presione el botón de retroceso en el cuadro de opción para eliminar el marcador seleccionado.

Teclas de método abreviado

Mientras edita, si presiona y mantiene estas secuencias de teclas se realizará la configuración y saltará a un

marcador. Presione todas las teclas a la vez para activar el atajo.

Establecer un marcador: presione On + Shift + [una tecla de número]

Saltar a un marcador: presione On + [una tecla de número]

1538 Ir a

ID 1538 · título localizado en HelpEs

Le permite ir a una línea específica del programa

1537 Sangría

ID 1537 · título localizado en HelpEs

Si se marca, al presionar ENTER durante la edición, se insertará automáticamente en la fuente del programa una cantidad de espacios similar a la de la línea anterior. Este es el comportamiento predeterminado del editor.

1535 Modo de Solo lectura

ID 1535 · título localizado en HelpEs

Pulse para alternar el indicador de 'solo lectura' del programa. El código fuente de un programa de solo lectura no se puede modificar en el editor hasta que el usuario desactive el estatus de solo lectura. Esto se puede usar para proteger un programa contra modificaciones accidentales.

1536 Encriptación

ID 1536 · título localizado en HelpEs

Cuando un programa está encriptado, su código fuente no es visible ni editable. No obstante, seguirá ejecutándose normalmente como un programa sin encriptación.

Cuando encripta un programa, el sistema le pedirá un nuevo nombre. Una vez que lo haya brindado, la HP Prime hará una copia del programa actual y lo guardará en un formato encriptado. El archivo de su programa original se mantendrá sin alteraciones. No descarte su programa original o la edición futura resultará imposible.

Este recurso está diseñado para brindar una mayor protección contra la modificación del programa. Evita que la mayoría de los usuarios vean o alteren su código fuente. Tome en cuenta que esto no evitará que un experto hábil en seguridad pueda desencriptar y alterar su programa.

1510 Catálogo de notas

ID 1510 · título localizado en HelpEs

Presione Shift 0 (notes) para abrir el Catálogo de nota.

Botones de menú:

- **[[Editar]]** abre la nota destacada para que la pueda editar en el Editor de notas
- **[[Nuevo]]** crea una nueva nota
- **[[Más]]** muestra más opciones de menú (guardar, renombrar, ordenar, eliminar y borrar) para la nota seleccionada
- **[[Enviar]]** envía la nota seleccionada a otra calculadora HP Prime

Nota: cuando sale del editor, este guarda los cambios automáticamente. Si desea guardar la versión original de su nota antes de realizar cambios, asegúrese de utilizar el botón Más en el Catálogo de notas y seleccionar Guardar.

1511 Editor de notas

ID 1511 · título localizado en HelpEs

El Editor de la nota es donde puede crear o modificar una nota.

Botones de menú:

- **[[Form.]]** muestra un menú de opciones de formato
- **[[Estilo]]** muestra un menú de opciones de estilo
- **[-▲ Página ▼-]** le permite desplazarse de una página a otra en una nota de varias páginas
- **[[●]]** pasa por los estilos de las viñetas
- **[[Insertar]]** toque para mostrar un menú de elementos que se pueden insertar

Pulse dos veces ALPHA para bloquear el desplazamiento alfa. Pulse de nuevo para volver a activarlo.

Puede copiar y pegar texto usando Shift View (Copy) y Shift Menu (Paste) respectivamente.

1514 Modo Examen

ID 1514 · título localizado en HelpEs

La calculadora HP Prime puede configurarse de forma precisa para un examen, con las funciones y características que desee desactivadas durante un periodo de tiempo establecido. La configuración de una calculadora HP Prime para un examen se denomina configuración del Modo Examen. Puede crear y guardar varias configuraciones del Modo Examen, cada una con su propio subconjunto de funciones desactivadas. Puede establecer cada configuración para un periodo de tiempo determinado, con o sin contraseña.

La configuración del Modo Examen será de interés principalmente para profesores, examinadores y supervisores que quieran garantizar que los estudiantes utilicen la calculadora de forma adecuada durante un examen.

El Modo Examen se puede configurar y activar en la página 3 de Configuración de Inicio. Puede acceder rápidamente a esta página manteniendo pulsada la tecla ON y pulsando Esc.

Configuraciones del Modo Examen

El campo Configuración permite seleccionar una configuración del Modo Examen. La HP Prime incluye el Modo básico, CSE Nederland y el Modo personalizado. También puede crear configuraciones adicionales basadas en el Modo personalizado.

Los botones de menú disponibles dependen de la configuración del Modo Examen seleccionada:

- [[Config]] permite ver y editar la configuración de examen seleccionada. Este botón está disponible para el Modo personalizado y las configuraciones creadas por el usuario.
- [[Selec.]] permite seleccionar una opción para el campo seleccionado actualmente.
- [-▲ Página 3/4 ▼-] permite volver a la página anterior (lado izquierdo del botón) o ir a la página siguiente (lado derecho del botón).
- [[Más]] muestra opciones para copiar o restablecer la configuración seleccionada. Este botón está disponible para el Modo personalizado y las configuraciones creadas por el usuario.
- [[Inicio]]/[[Enviar]] permite activar el Modo Examen o enviar la configuración seleccionada a otra HP Prime.

Tenga en cuenta que [[Enviar]] solo aparece cuando la calculadora HP Prime está conectada a otra HP Prime mediante el cable USB, y únicamente en los modelos HP Prime que admiten conectividad directa entre calculadoras.

Modo básico

El Modo básico es una configuración predefinida del Modo Examen. Seleccione Modo básico en el campo Configuración para utilizarlo.

Está disponible la siguiente opción:

- Ángulo predeter.: seleccione Sin cambios para utilizar la configuración actual de la calculadora, o seleccione Radianes, Grados o Gradianes como medida de ángulo que se utilizará durante el Modo Examen.

En el Modo básico:

- La memoria de la HP Prime permanece oculta mientras el Modo Examen está activo y se restaura cuando el Modo Examen finaliza.
- El LED verde de la parte superior de la calculadora parpadea.

Toque [[Inicio]] para activar el Modo básico.

CSE Nederland

CSE Nederland es una configuración predefinida y protegida del Modo Examen. Sus ajustes no se pueden modificar.

Seleccione CSE Nederland en el campo Configuración para utilizar esta configuración.

Modo personalizado

El Modo personalizado es una configuración del Modo Examen que se puede modificar. Puede cambiar sus ajustes y utilizarlo como base para crear configuraciones adicionales. Las configuraciones creadas a partir del Modo personalizado también se pueden modificar y copiar.

Los siguientes campos están disponibles para el Modo personalizado y las configuraciones creadas por el usuario:

- Tiempo de espera: permite seleccionar una duración para el Modo Examen, es decir, el periodo durante el cual las funciones estarán desactivadas.
- Ángulo predeter.: seleccione Sin cambios para utilizar la configuración actual de la calculadora, o seleccione Radianes, Grados o Gradianes como medida de ángulo que deberán utilizar todas las calculadoras cliente.

- Contraseña: código que permite desactivar el Modo Examen antes de que finalice el periodo de espera.
- Memoria: permite Mantener, Eliminar, Ocultar o Mantener y eliminar los cambios realizados en la memoria durante el Modo Examen.
- Parpadeo de LED: hace que el LED de todas las calculadoras cliente siga un patrón secuencial fijo.
- Código de seguridad: active esta opción para solicitar un código de seguridad al iniciar el Modo Examen; el código determina el patrón de parpadeo del LED.

Toque [[Config]] para ver o modificar las funciones desactivadas por la configuración seleccionada.

Finalización del Modo Examen

Independientemente de la configuración seleccionada, el Modo Examen finaliza cuando la calculadora se conecta a un PC o a otra HP Prime mediante el cable micro-USB.

1515 Configuración de modo Examen

ID 1515 · título localizado en HelpEs

En la página 3 de la pantalla Configuración de Inicio, toque [[Config.]]. Aparece la pantalla Configuración del Modo Examen, que muestra un árbol de características que se pueden desactivar. Cada entrada dispone de una casilla de verificación a la izquierda. Toque una entrada para seleccionarla y, a continuación, toque su casilla de verificación o el botón de menú [[✓]] para desactivarla.

Algunas entradas tienen un signo más (+) a la izquierda de sus casillas de verificación. Estas entradas representan categorías. Puede desactivar una categoría completa marcando su casilla de verificación, o tocar el signo más (+) para expandir la categoría y seleccionar características individuales.

Para seleccionar las características que desea desactivar:

- 1. Seleccione las características que desea desactivar. Asegúrese de que las características que desea mantener disponibles no estén seleccionadas.
- 2. Toque [[OK]] para volver a la pantalla de configuración del Modo Examen.

1516 Creación de configuraciones nuevas

ID 1516 · título localizado en HelpEs

Además de modificar la configuración del Modo personalizado, puede crear nuevas configuraciones de Modo Examen basadas en ella. A continuación, puede seleccionar la configuración que desee antes de activar el Modo Examen.

Para crear una nueva configuración de Modo Examen:

- 1. En la página 3 de Configuración de Inicio, seleccione el Modo personalizado o una configuración creada anteriormente que desee utilizar como base para la nueva configuración. Si todavía no ha creado ninguna configuración, el Modo personalizado es la única opción configurable disponible.
- 2. Toque [[Más]] y seleccione Copiar. Aparece la pantalla Nueva configuración de Modo Examen.
- 3. Escriba un nombre para la nueva configuración.
- 4. Toque [[OK]] dos veces.
- 5. Toque [[Config.]].
- 6. Seleccione las funciones que desea desactivar. Asegúrese de que las funciones que desea mantener disponibles no estén seleccionadas.
- 7. Toque [[OK]].

La nueva configuración se añade a la lista desplegable Configuración y puede seleccionarse antes de activar el Modo Examen.

1517 Activación del Modo Examen

ID 1517 · título localizado en HelpEs

El Modo Examen se puede activar en la página 3 de Configuración de Inicio. Puede acceder rápidamente a esta página manteniendo pulsada la tecla On y pulsando Esc.

Para activar el Modo Examen:

- 1. En la página 3 de Configuración de Inicio, seleccione la configuración que desea activar.

- 2. Toque **[[Inicio]]**. Aparece una pantalla de resumen que indica que está a punto de activar el Modo Examen. Deslice el candado para activar el Modo Examen, o toque **[[Cancelar]]** para volver a la página 3 de Configuración de Inicio.
- 3. Según la configuración del Modo Examen seleccionada, es posible que se le solicite introducir un código de seguridad proporcionado por el supervisor.

Siempre puede salir del Modo Examen conectando la HP Prime a un PC mediante el cable USB o a otra HP Prime mediante el cable micro-USB de conexión directa entre calculadoras.

Según la configuración del Modo Examen seleccionada, también puede salir del Modo Examen:

- introduciendo la contraseña en la pantalla de configuración del Modo Examen y tocando **[[OK]]**
- esperando a que finalice el periodo de espera

También puede activar la configuración actual del Modo Examen en otra HP Prime conectada a la suya mediante el cable USB, siempre que ambas calculadoras admitan conectividad directa entre calculadoras. Cuando las calculadoras están conectadas, el botón de menú **[[Inicio]]** se sustituye por **[[Enviar]]**. Toque **[[Enviar]]** para instalar y activar la configuración actual del Modo Examen en la HP Prime conectada.

También puede instalar y activar una configuración del Modo Examen en una o más calculadoras HP Prime mediante el Kit de conectividad. Las calculadoras pueden comunicarse con el Kit de conectividad mediante un cable USB o, en los modelos HP Prime que admiten conectividad inalámbrica, mediante el módulo inalámbrico opcional. Consulte la Guía de usuario del Kit de conectividad para obtener más información.

1512 Mensajería

ID 1512 · título localizado en HelpEs

Cuando esté conectado a una PC de forma inalámbrica o a través de USB, la PC puede enviar mensajes a la calculadora. También puede enviar mensajes a la PC. Consulte la Guía de usuario del Kit de conectividad de HP Prime para obtener más información.

Descarte el mensaje pulsando **[[OK]]**; o bien, toque **[[Reproducir]]**, introduzca su mensaje y toque **[[OK]]** para enviarlo.

1513 Encuesta y cuestionario

ID 1513 · título localizado en HelpEs

La funcionalidad de Encuesta y cuestionario permite la comunicación entre una computadora y cualquier cantidad de calculadoras. Posibilita que los profesores se comuniquen de forma inalámbrica con las calculadoras de los alumnos, les envíen preguntas y reciban respuestas. Las respuestas se agregan y se muestran. Estos recursos permiten la evaluación formativa, así como la participación activa de los estudiantes en las actividades de clase.

La función de Encuesta y cuestionario requiere el Kit de conectividad HP Prime. Consulte a la Guía del usuario que acompaña el Kit de conectividad HP Prime para obtener instrucciones.

Apéndice A. Entradas no visibles en el árbol de interfaz

Estas entradas existen en los recursos analizados, pero no forman parte del árbol visible reconstruido para esta versión del Virtual Calculator. Se incluyen para mantener la cobertura completa.

0004 MicroPython

ID 4 · entrada preservada fuera del árbol visible

The following copyright and license information applies to, and only to, the MicroPython software.

MicroPython license information

The MIT License (MIT)

Copyright (c) 2013-2017 Damien P. George, and others

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

0233 Vista de gráfico

ID 233 · entrada preservada fuera del árbol visible

La Vista de gráfico de Hoja de cálculo muestra las representaciones gráficas de una o más series de datos puntuales introducidos en la vista numérica de la hoja de cálculo.

Se usa en la Vista simbólica para configurar el contenido de la Vista de gráfico.

Use los botones del menú [-▲ Página ▼-] para seleccionar el gráfico que quiere analizar.

En la Vista de gráfico, siempre se selecciona uno de los puntos de datos. Su posición en el gráfico se indica con una pequeña cruz y la parte inferior de la pantalla proporciona la fuente de información de datos.

Use las teclas de flecha hacia arriba/hacia abajo para cambiar la serie seleccionada.

Utilice las flechas izquierda/derecha para trazar/seleccionar el punto siguiente/anterior de una serie de números mostrada.

Toque la pantalla para cambiar la selección.

0234 Vista simbólica

ID 234 · entrada preservada fuera del árbol visible

La Vista simbólica de la aplicación Hoja de cálculo le permite configurar los gráficos que desea ver en la vista de gráfico.

Esta vista es una serie de cuadros de diálogo, uno por cada gráfico en la vista de gráfico.

Utilice el botón Nuevo para agregar un gráfico (verá que el número de página en el cuadro de diálogo del menú página arriba/abajo aumenta).

Utilice el botón Eliminar para borrar la definición de gráfico actual.

Utilice los botones del menú arriba/abajo para desplazarse por la lista de gráficos.

Para cada gráfico, puede especificar:

- Un nombre que se mostrará en la parte superior de la vista de gráfico.

- Una fuente de la hoja de cálculo (A1: D4), por ejemplo.
- Un Tipo: barras, líneas, área, diagrama de torta o gráfico de dispersión
- Un subtipo: agrupado, apilado o el 100 % total para Barras, Líneas y Área. Separada o Hanoi por el diagrama de torta y líneas, líneas y puntos o puntos para los gráficos de dispersión.
- Puede especificar hasta 5 colores que se utilizarán conforme al código de color de la serie.

Entonces, por ejemplo, en la vista numérica:

- Seleccione A1 a D4.
- Escriba [Ejemplo gráfico E1] ENTER (ver ejemplo)
- Presione Symb e ingrese A1: D4 como fuente.
- Presione Plot para ver el gráfico.

0447 Aplicación Explorador cuadrático

ID 447 · entrada preservada fuera del árbol visible

La aplicación Explor. cuadrático sirve para estudiar el comportamiento de una parábola, ya sea manipulando la ecuación y observando el cambio en el gráfico, o bien manipulando el gráfico y observando el cambio en la ecuación.

Para iniciar la aplicación Explor. cuadrático, vaya a la biblioteca de aplicaciones y pulse el icono de la aplicación Explor. cuadrático. También puede utilizar la rueda para seleccionar el icono de la aplicación Explor. cuadrático, luego pulse [[Inicio]] o presione Enter para iniciar la aplicación.

0448 Vista de gráfico del Explor. cuadrático

ID 448 · entrada preservada fuera del árbol visible

El Explor. cuadrático se abre en la Vista de gráfico con el modo Gráf. activo. La parte izquierda de la pantalla muestra el gráfico de una función cuadrática. La parte derecha muestra la forma general de la ecuación que se explora en la parte superior. Debajo de ella se encuentra la ecuación actual de dicha fórmula.

Las teclas que se utilizan para controlar la aplicación aparecen debajo de la ecuación. En la parte inferior derecha se muestra la ecuación en su fórmula estándar, el valor del discriminante y la raíz o raíces de la función cuadrática.

Las teclas de menú son:

- [[Ec]] cuando está activo, utiliza la rueda para cambiar entre parámetros de la ecuación y modificar sus valores.
- [[GRÁF]] cuando está activa, utiliza la rueda para manipular el gráfico.
- [[Inc n]] permite modificar el incremento en el que cambian los valores de parámetros
- [[Niv n]] permite cambiar entre diversas fórmulas de funciones cuadráticas
- [[Prueba]] permite acceder a la Vista de prueba de la aplicación

Presione Shift Esc (Clear) para reiniciar la aplicación en cualquier momento.

0449 Modo de gráfico del Explorador cuadr.

ID 449 · entrada preservada fuera del árbol visible

En el modo Gráf., puede trasladar y dilatar el gráfico directamente y ver los efectos resultantes en los valores de los parámetros de la ecuación a medida que manipula el gráfico.

Las teclas activas son:

- Utilice la rueda para trasladar el gráfico
- Utilice las teclas + y - para ampliar el gráfico
- Utilice la tecla +/- para mostrar el gráfico en el eje X

Presione la tecla de menú Gráf. para volver a este modo en cualquier momento.

0450 Modo de ecuación del Explor. cuadrático

ID 450 · entrada preservada fuera del árbol visible

Presione la tecla de menú EC para acceder al modo ecuación. En este modo, puede cambiar los parámetros de la ecuación directamente y observar el efecto sobre la forma del gráfico.

Las teclas activas son:

- Utilice la rueda hacia la izquierda/derecha para seleccionar un parámetro
- Utilice la rueda hacia arriba/abajo para aumentar y disminuir el valor del parámetro seleccionado
- Utilice las teclas +/- para cambiar el signo del parámetro a

Presione la tecla del menú Incr n para alternar entre diferentes valores de incremento.

Presione la tecla de menú Ec para volver a este modo en cualquier momento.

0451 Modo de prueba del Explor. cuadrático

ID 451 · entrada preservada fuera del árbol visible

Presione la tecla de menú [[Prueba]] para entrar en el modo de prueba. En el modo de prueba, puede probar su habilidad para hacer coincidir una ecuación con el gráfico que se muestra. El modo de prueba es como el modo de ecuación ya que utiliza la rueda para seleccionar y cambiar el valor de cada parámetro en la ecuación para que coincida con el gráfico que se muestra. Cuando haya terminado, presione la tecla de menú [[Comprobar]] para verificar si su respuesta es correcta.

Las teclas de menú son:

- Niv n: permite seleccionar uno de los cuatro niveles de dificultad
- Marcar: permite comprobar la respuesta actual con respecto a la respuesta correcta
- Resp.: muestra la respuesta correcta
- Final.: permite volver a la Vista de gráfico

0452 Aplicación Explor. trigonom

ID 452 · entrada preservada fuera del árbol visible

La aplicación Explor. trigonom. sirve para estudiar el comportamiento del gráfico de $y=a*\sin(bx+c)+d$ o $y=a*\cos(bx+c)+d$ a medida que los valores de a, b, c y d cambian, ya sea al manipular la ecuación y observar el cambio en el gráfico, o bien al manipular el gráfico y observar el cambio en la ecuación.

Para iniciar la aplicación Explor. trigonom., vaya a la biblioteca de aplicaciones y pulse el icono de la aplicación Explor. trigonom. También puede utilizar la rueda para seleccionar el icono de la aplicación Explor. trigonom., luego pulse [[Inicio]] o presione Enter para iniciar la aplicación.

0453 Vista de gráfico Explor. trigonom.

ID 453 · entrada preservada fuera del árbol visible

La aplicación Explor. trigonom. se abre en la Vista de gráfico con el modo Gráf activo. La ecuación del gráfico actual se muestra en la parte superior de la pantalla con el gráfico debajo de ella.

Los botones del menú son:

- Gráf/Ec: permite cambiar entre los modos Gráf y Ec.
- SEN/COS: permite cambiar entre la exploración de gráficos de senos y de cosenos
- Rad/Grad: permite cambiar entre la medida del ángulo de radianes y grados para x
- Orig/Extr: permite cambiar entre trasladar el gráfico (Orig) y dilatarlo (Extr) utilizando la rueda
- π/n (o n): permite modificar el incremento en el que cambian los valores de los parámetros
- Prueba: permite acceder a la vista de prueba de la aplicación

Presione Shift Retroceso (Del) y, a continuación, toque [[OK]] para restablecer la aplicación en cualquier momento.

0454 Modo gráfico del Explorador de trigonometría

ID 454 · entrada preservada fuera del árbol visible

En el modo Gráfico, puede trasladar y dilatar el gráfico directamente y ver los efectos resultantes respecto de los valores de los parámetros de la ecuación mientras manipula el gráfico. El comportamiento de la rueda depende del ajuste del botón de menú Orig/Extr.

Las teclas activas son:

- Utilice la rueda para trasladar (Orig) y dilatar (Extr) el gráfico, de manera vertical u horizontal
- Utilice la tecla +/- para mostrar el gráfico en el eje X

El incremento en el que cambian los parámetros a , b y d siempre es 0.1; el incremento para el parámetro c depende de la configuración del botón de menú Rad/Grad y el botón de menú π/n (o n). Toque Gráf/Ec para volver a este modo en cualquier momento.

0455 Modo EC del Explor. trigonom.

ID 455 · entrada preservada fuera del árbol visible

Toque Gráf/Ec para cambiar el modo Ec. En este modo, puede cambiar los parámetros de la ecuación directamente y observar el efecto sobre la forma del gráfico.

Las teclas activas son:

- Utilice la rueda hacia la izquierda/derecha para seleccionar un parámetro
- Utilice la rueda hacia arriba/abajo para aumentar y disminuir el valor del parámetro seleccionado.
- Utilice las teclas +/- para cambiar el signo del parámetro a

Los incrementos en el modo Ec se comportan igual que en el modo Gráf. Toque Gráf/Ec para volver a este modo en cualquier momento.

0456 Modo prueba del explorador de trigonometría

ID 456 · entrada preservada fuera del árbol visible

Pulse Prueba para introducir la vista Prueba en la aplicación. En la vista Prueba, puede probar sus habilidades para asociar una ecuación con el gráfico mostrado. La vista Prueba es similar al modo Ec en el que usa la rueda para seleccionar y cambiar el valor de cada parámetro en la ecuación para coincidir con el gráfico mostrado. Cuando haya terminado, pulse Verificar para revisar si su respuesta está correcta.

Los botones del menú son:

- Rad/Grad: permite seleccionar la medida del ángulo para la prueba
- π/n (o n): permite seleccionar el incremento para la prueba
- Nivn: permite seleccionar el nivel para la prueba
- Marcar: permite comprobar la respuesta actual con respecto a la respuesta correcta
- Resp.: muestra la respuesta correcta
- Final.: permite volver a la Vista de gráfico

0498 Aplicación Explorador lineal

ID 498 · entrada preservada fuera del árbol visible

La aplicación Explorador lineal es un micromundo diseñado para explorar pendientes e interceptaciones de funciones lineales de la forma $y=ax$ e $y=ax+b$. Esta aplicación sirve para estudiar las características del gráfico a medida que los valores de a y b cambian, ya sea manipulando la ecuación y observando el cambio en el gráfico, o bien manipulando el gráfico y observando el cambio en la ecuación.

Para iniciar la aplicación Explorador lineal, vaya a la biblioteca de aplicaciones y pulse el icono de la aplicación Explorador lineal. También puede utilizar la rueda para seleccionar el icono de la aplicación Explorador lineal, luego pulse [[Inicio]] o presione Enter para iniciar la aplicación.

0499 Vista de gráfico de Explorador lineal

ID 499 · entrada preservada fuera del árbol visible

El Explorador lineal se abre en la Vista de gráfico con el modo GRÁF activo. La parte izquierda de la pantalla muestra el gráfico de una función lineal. La parte derecha muestra la forma general de la ecuación que se explora en la parte superior. Debajo de ella se encuentra la ecuación actual de dicha fórmula. Las teclas que se utilizan para controlar la aplicación aparecen debajo de la ecuación y las interceptaciones de X e Y se muestran en la parte inferior.

Las teclas de menú son:

- EC: cuando está activa, utiliza la rueda para alternar entre los parámetros de la ecuación y cambiar sus valores
- GRÁF.: cuando está activa, utiliza la rueda para manipular el gráfico
- INCn: permite modificar el incremento en el que cambian los valores de parámetros
- NIVn: permite cambiar entre ecuaciones lineales $y=ax$ e $y=ax+b$.
- PRUEBA: permite acceder a la vista de prueba de la aplicación

Pulse Shift + Retroceso (Del) y luego pulse la tecla OK del menú para restablecer la aplicación en cualquier momento.

0500 Modo Gráfico del explorador lineal

ID 500 · entrada preservada fuera del árbol visible

En el modo GRÁF, la forma del gráfico se cambia directamente y se observa el efecto sobre los parámetros de la ecuación.

Las teclas activas son:

- Utilice la rueda hacia arriba/abajo para trasladar el gráfico (cambia la intercepción)
- Utilice las teclas + y - para ampliar el gráfico (cambia la pendiente)
- Utilice la tecla +/- para reflejar el gráfico sobre el eje Y (cambia el signo de la pendiente)

Si la tecla de menú NIV n está ajustada a NIV 1 (para $y=ax$), entonces la rueda hacia arriba/abajo no funciona y esto se refleja en el conjunto de teclas activas.

Pulse la tecla de menú GRÁF para volver a este modo en cualquier momento.

0501 Modo EC de explorador lineal

ID 501 · entrada preservada fuera del árbol visible

Pulse la tecla de menú EC para acceder al modo EC. En este modo, los parámetros de la ecuación se cambian directamente y se observa el efecto sobre la forma del gráfico.

Las teclas activas son:

- Utilice la rueda hacia la izquierda/derecha para seleccionar un parámetro
- Utilice la rueda hacia arriba/abajo para aumentar y disminuir el valor del parámetro.
- Utilice las teclas +/- para cambiar el signo del parámetro

Pulse la tecla de menú INCn para alternar entre los diferentes valores de incremento.

Pulse la tecla de menú EC para regresar a este modo en cualquier momento.

0502 Modo de prueba de Explorador lineal

ID 502 · entrada preservada fuera del árbol visible

Presione el menú PRUEBA para acceder al modo de prueba de la aplicación. En el modo de prueba, puede comprobar su habilidad para hacer coincidir una ecuación con la gráfica que se muestra. La Vista de prueba es como el modo de ecuación ya que se usa la rueda para seleccionar y cambiar el valor de cada parámetro de la ecuación. El objetivo es intentar hacer coincidir la gráfica que se muestra. Cuando haya terminado, presione la tecla de menú Compr. para comprobar si la respuesta es correcta.

Las teclas de menú son:

- NIVn: permite seleccionar $y=ax$ (NIV 1) o $y=ax+b$ (NIV 2) para la prueba
- MARCAR: permite comprobar la respuesta actual con respecto a la respuesta correcta
- RESP.: muestra la respuesta correcta
- FINAL.: permite volver a la Vista de gráfico

0503 Soluc. para pendiente

ID 503 · entrada preservada fuera del árbol visible

Sintaxis

SolveForSlope(x1, y1, x2, y2)

Soluc. para pendiente. Toma como entrada las coordenadas de dos puntos (x1,y1) and (x2,y2) y devuelve la pendiente de la línea que contiene los dos puntos.

0504 Soluc. para interceptación Y

ID 504 · entrada preservada fuera del árbol visible

Sintaxis

SolveForYIntercept(x, y, m)

Toma como entrada las coordenadas de un punto (x, y), y una pendiente m, y devuelve la interceptación y de la línea con la pendiente dada que contiene el punto dado.

0889 Airy_Ai — Devuelve el valor Ai de la solución función Airy de $w''-xw=0$.

ID 889 · entrada preservada fuera del árbol visible

Referencia traducida

Identificador técnico: Airy_Ai

Explicación española: Devuelve el valor Ai de la solución función Airy de $w''-xw=0$.

Sintaxis

Airy_Ai(x)

Devuelve el valor Ai de la solución función Airy de $w''-xw=0$.

0890 Airy_Bi — Devuelve el valor Bi de la solución función Airy de $w''-xw=0$.

ID 890 · entrada preservada fuera del árbol visible

Referencia traducida

Identificador técnico: Airy_Bi

Explicación española: Devuelve el valor Bi de la solución función Airy de $w''-xw=0$.

Sintaxis

Airy_Bi(x)

Devuelve el valor Bi de la solución función Airy de $w''-xw=0$.

1246 Descomposición de valor único

ID 1246 · entrada preservada fuera del árbol visible

Sintaxis

SVD(Mtrx(A))

Factoriza una matriz $m \times n$ en dos matrices y un vector: $\{[m \times m \text{ ortogonal}], [m \times 1 \text{ vector de valores singulares}], [n \times n \text{ ortogonal}]\}$.

1522 No se ha encontrado ayuda

ID 1522 · entrada preservada fuera del árbol visible

Disculpe, la ayuda en línea para este elemento no está disponible.

Para navegar por todo el sistema de ayuda, pulse [\[\[Árbol\]\]](#). A continuación, puede utilizar la rueda para navegar por el sistema y abrir los temas de ayuda.

Si todavía no puede encontrar el tema que desea, consulte a la Guía del usuario.

Pulse [\[\[OK\]\]](#) para salir de esta página.

1640 Fuente del formato

ID 1640 · entrada preservada fuera del árbol visible

Cuando se pulsa, el texto seleccionado en el editor del programa se reformateará automáticamente basado en el contenido. Esto hará que elementos emparejados como BEGIN y END se organicen en la misma columna y realicen una limpieza de código similar.

1680 Vista de prueba del explorador

ID 1680 · entrada preservada fuera del árbol visible

Pulse la tecla de menú [[Prueba]] para entrar en el modo de prueba. En el modo Prueba, prueba su habilidad para hacer coincidir una ecuación con el gráfico que se muestra. El modo de prueba muestra el gráfico de una función elegida al azar de la forma determinada por su elección en la Vista simbólica, así como su elección de nivel. Pulse Niv1 o Niv2 para cambiar el nivel. El Modo de prueba es como el modo de ecuación ya que edita el valor de cada parámetro en la ecuación para tratar de que coincida con el gráfico que se muestra. Cuando esté listo, pulse la tecla de menú [[Compr.]] para ver si su ecuación coincide correctamente con el gráfico. Pulse Resp. para ver la respuesta correcta y pulse Final. para salir del modo de prueba.

En el modo de prueba, las teclas de menú comunes son:

- [[Niv n]] permite seleccionar uno de los cuatro niveles de dificultad
- [[Marcar]] permite comprobar la respuesta actual con respecto a la respuesta correcta
- [[Resp.]] muestra la respuesta correcta
- [[Final.]] permite volver a la Vista de gráfico

1725 insert — Devuelve el vector o la cadena donde se ha insertado el elemento `b` en el índice `n`.

ID 1725 · entrada preservada fuera del árbol visible

Referencia traducida

Identificador técnico: insert

Explicación española: Devuelve el vector o la cadena donde se ha insertado el elemento `b` en el índice `n`.

Sintaxis

```
insert(vector, n, b)
insert(string, n, b)
```

Devuelve el vector o la cadena donde se ha insertado el elemento b en el índice n.

Apéndice B. Notas técnicas de reconstrucción

La organización se obtuvo de la tabla de entradas de `HelpEs.hpresource/HelpEn.hpresource`, del flujo de cuerpos de ayuda y de la jerarquía serializada utilizada por `CHelpTreeControl` en el ejecutable analizado. Las relaciones padre-hijo se conservan tal como las procesa la interfaz.

Offset del árbol: 0xD6D820

Fin del árbol: 0xD7111C

Índice alfabético de comandos, funciones y variables

A

A
a2q
a_b[]c
AAngle
AAS
abcuv
about
ABS
abs (builtins)
abs (linalg)
abscissa
Acceleration
Accrued
AComplex
ACOS
acos (math)
acos2asin
acos2atan
ACOSH
acosh (math)
ACOT
ACSC
add (builtins.set)
add (linalg)
ADDCOL
additionally
ADDDROW
ADigits
adjoint_matrix
Advanced
affix
AFiles
AFilesB
AFormat
Airy_Ai
Airy_Bi
Algebra
algvar
all (builtins)
ALOG
alog10
Alpha
ALPHA
AltHyp
altitude
AMORT
and
AND
angle
Angle
AngleA
angleat
AngleB
AngleC
ANote
Anova0neWay
Ans
any (builtins)
append
append (array.array)
append (builtins bytearray)
append (builtins.list)
append (ucollections)
apply
apply (linalg)
approx
Apps
AProgram
arange (linalg)
ARC
arc
arc (hprime)
arc_c (hprime)
ARC_P
arcLen

AREA
 area
 Area
 areaat
 ARG
 argv (sys)
 arit
 Arithmetic
 array
 array class (array)
 ASA
 ASC
 asc (arit)
 ASEC
 ASIN
 asin (math)
 asin2acos
 asin2atan
 ASINH
 asinh (math)
 assert
 assume
 ATAN
 atan (math)
 atan2
 atan2 (math)
 atan2acos
 atan2asin
 ATANH
 atanh (math)
 atrig2ln
 AVars
 AVERAGE
 Axes
 axis (matplotlib)

B

B
 Backspace
 Backup
 Backups
 barplot (matplotlib)
 barycenter
 Base
 Basic
 basis
 BEG
 BEGIN END
 bernoulli
 Beta
 betad
 betad_cdf
 betad_icdf
 bin
 bin (builtins)
 BINOMIAL
 BINOMIAL_CDF
 BINOMIAL_ICDF
 bisector
 BITAND
 BITNOT
 BITOR
 Bits
 BITS_L
 BITS_R
 BITXOR
 BlackScholes
 BLIT
 blit (hprime)
 blit_c (hprime)
 BLIT_P
 Block
 blockmatrix
 Bond360
 BondPrice
 BondYield
 Bookmark
 bool
 bool class (builtins)
 bounded_function
 boxplot (matplotlib)
 Branch
 BREAK

break (with loop)
breakpoint
BrkEvCost
BrkEvFixed
BrkEvPrice
BrkEvProfit
BrkEvQuant
BSCall
BSPut
builtins
bytearray class (builtins)
byteorder (sys)
bytes class (builtins)
BytesIO class (uio)
B→R

C

C
cloc2
clop2
calcsize (ustruct)
Calculus
callable (builtins)
CallPrice
canonical_form
Cartesian
cas
CAS
CASE
caseval (cas)
CashFlowFMRR
CashFlowIRR
CashFlowMIRR
CashFlowNFV
CashFlowNPV
CashFlowNUS
CashFlowPB
CashFlowTotal
cat
cauchy
cauchy_cdf
cauchy_icdf
ceil (math)
CEILING
Cell References and Cell
CellHasData
CellName
center
center (builtins.bytes)
center (builtins.str)
cFactor
CFData
CFPYR
Change
changebase
ChangeCost
ChangeNew
ChangeOld
ChangePrice
CHAR
char (arit)
charpoly
CHECK
Chemistry
Chi2GOF
Chi2TwoWay
chinrem
CHISQUARE
CHISQUARE_CDF
CHISQUARE_ICDF
chisquaret
choice
choice (urandom)
cholesky
CHOOSE
CHOOSE (Spreadsheet)
CHOOSEDATE
chr (builtins)
chrem
Ci
circle
circle (hpprime)
circle_c (hpprime)

```

circumcircle
class (class block)
classmethod (builtins)
Clear
clear (builtins.dict)
clear (builtins.list)
clear (builtins.set)
clear (ucollections.OrderedDict)
clear_screen (graphic)
ClearCell
clf (matplotlib)
close (uio.BytesIO)
close (uio.FileIO)
close (uio.StringIO)
close (uio.TextIOWrapper)
cmath
CoefDet
coefDet
coeff
Col
col
colDim
collect
collect (gc)
COLNORM
colnorm
colspace
ColWidth
COMB
comb
comDenom
common_perpendicular
companion
compare
compile (builtins)
compile (ure)
complex
Complex
complex class (builtins)
complexroot
CONCAT
concat
COND
Conf
ConfT1mean
ConfT2mean
ConfZ1mean
ConfZ1prop
ConfZ2mean
ConfZ2prop
conic
CONJ
conj
conj (linalg)
CONST
const (micropython)
Constants
contains
content
CONTINUE
continue (with loop)
ContribList
ContribMat
CONVERT
convert
convexhull
CoordinateConvert
coordinates
Copy
copy (builtins.dict)
copy (builtins.frozens)
copy (builtins.list)
copy (builtins.set)
copy (ucollections.OrderedDict)
copysign (math)
CopyVar
Corr
corr
correlation
COS
cos (cmath)
cos (math)
cos2sintan
COSH
cosh (math)

```

Cost
 CostAsset
 COT
 count
 count (builtins.bytes)
 count (builtins.list)
 count (builtins.str)
 count (builtins.tuple)
 count_eq
 count_inf
 count_sup
 covariance
 covariance_correlation
 cpartfrac
 CpnPer
 CPYR
 crationalroot
 Create
 CritScore
 CritVal1
 CritVal2
 CROSS
 cross (linalg)
 CSC
 cSolve
 cumSum
 Cumulative
 curl
 Cursor
 Curve
 curve
 cycle2perm
 cycleinv
 cycles2perm
 cyclotomic
 cZeros
 C→PX

D

D
 DataType
 Date
 Date360
 DATEADD
 DateDays
 DateDiff
 DateOne
 DateTwo
 DAYOFWEEK
 DDAYS
 DEBUG
 DEBUG (ure)
 decode (builtins.bytes)
 def (function block)
 Define
 degree
 degrees (math)
 Del
 DelAFiles
 DelApp
 delattr (builtins)
 DelAVars
 DelBackup
 DELCOL
 delcols
 deleter (builtins.property)
 DelHVars
 DelInstruction
 DELROW
 delrows
 deltalist
 denom
 Density
 Depreciate
 deque class (ucollections)
 desolve
 DET
 det (linalg)
 DF
 dfc
 dfc2f
 DFerr
 diag

```

dict class (builtins)
diff
DIFFERENCE
difference (builtins.frozens)
difference (builtins.set)
difference_update (builtins.set)
Differential
digest (uhashlib.sha256)
DIM
DIMGROB
dimgrob (hprime)
dimgrob_c (hprime)
DIMGROB_P
dir (builtins)
Dirac
disable (gc)
discard (builtins.set)
DiscPayback
distance
distance2
distanceat
divergence
Dividend
divis
Division
division_point
divmod (builtins)
divpc
Do1VStats
Do2VStats
DoInference
domain
DoSolve
DOT
dot (linalg)
draw_arc (graphic)
draw_circle (graphic)
draw_filled_arc (graphic)
draw_filled_circle (graphic)
draw_filled_polygon (graphic)
draw_filled_rectangle (graphic)
draw_line (graphic)
draw_pixel (graphic)
draw_polygon (graphic)
draw_string
draw_string (graphic)
Drawing
DRAWMENU
DrawSlp

```

E

```

E
e
e (cmath)
e (math)
e^x
EDITLIST
EDITMAT
EEX
EffInt
egcd
Ei
eig (linalg)
EIGENVAL
eigenvals
eigenvecs
eigenvecs (linalg)
EIGENVV
eigVc
eigVl
Electrical
element
elif (if-elif-else block)
ellipse
Ellipsis (builtins)
else (if-else block)
else (with for loop)
enable (gc)
encode (builtins.str)
Encryption
END
endswith (builtins.bytes)
endswith (builtins.str)

```

- Energy
- Enter
- Entering Content
- Entry
- enumerate (builtins)
- EQ
- equal
- equation
- equilateral_triangle
- erf
- erf (math)
- erfc
- erfc (math)
- error
- Esc
- euler
- euler (arit)
- euler_gamma
- EVAL
- eval
- Eval
- eval (builtins)
- eval (hpprime)
- evalb
- evalc
- evalf
- EVALLIST
- even
- exact
- exbisector
- exc_info (sys)
- Exceptions
- excircle
- exec (builtins)
- EXECON
- exit (sys)
- EXP
- exp (cmath)
- exp (math)
- exp2pow
- exp2trig
- expand
- expexpand
- ExpList
- EXPM1
- expm1 (math)
- ExpMat
- exponential
- exponential_cdf
- exponential_icdf
- exponential_regression
- EXPORT
- EXPR
- extend
- extend (array.array)
- extend (builtins bytearray)
- extend (builtins.list)
- Extract
- extract_measure
- Extremum
- EXTREMUM
- ezgcd

F

- F
- f2nd
- fabs (math)
- factor
- factor_xn
- FactorDepr
- factorial
- Factorize
- factors
- fcoeff
- fft
- fft (linalg)
- FileIO class (uio)
- fileno (uio.FileIO)
- fileno (uio.TextIOWrapper)
- FILLPOLY
- fillpoly (hpprime)
- fillpoly_c (hpprime)
- FILLPOLY_P

```

fillrect (hprime)
fillrect_c (hprime)
filter (builtins)
find (builtins.bytes)
find (builtins.str)
FinType
FirstAsset
FirstDateAsset
FISHER
FISHER_CDF
FISHER_ICDF
FixedCost
float
float class (builtins)
FLOOR
floor (math)
flush (uio.BytesIO)
flush (uio.FileIO)
flush (uio.StringIO)
flush (uio.TextIOWrapper)
fMax
fMin
fmod (math)
FMRR
FNROOT
for (range loop)
for (sequence loop)
FOR FROM TO DO END
Force
format
format (builtins.bytes)
format (builtins.str)
fourier_an
fourier_bn
fourier_cn
FP
fPart
fracmod
FREEZE
frexp
frexp (math)
from_bytes (builtins.int)
fromkeys (builtins.dict)
fromkeys (ucollections.OrderedDict)
froot
frozenset class (builtins)
fsolve
function_diff
FV
Fval

```

G

```

G
Gamma
gamma (math)
gammad
gammad_cdf
gammad_icdf
gauss
gbasis
gc
gcd
gcd (arit)
GEOMETRIC
geometric_cdf
geometric_icdf
GET
get (builtins.dict)
get (ucollections.OrderedDict)
get_cartesian (hprime)
get_key (cas)
get_pixel (graphic)
getattr (builtins)
GETBASE
GETBITS
GETKEY
GETPIX
getpix (hprime)
getpix_c (hprime)
GETPIX_P
getrandbits (urandom)
getter (builtins.property)
getvalue (uio.BytesIO)

```

```

getvalue (uio.StringIO)
GF
global
globals (builtins)
grad
gramschmidt
graphic
greduce
grid (matplotlib)
GridDots
GridLines
GROBH
grobh (hprime)
grobh_c (hprime)
GROBH_P
GROBW
grobw (hprime)
grobw_c (hprime)
GROBW_P
groupermu
GSize

```

H

```

H
hadamard
half_line
halftan
halftan_hyp2exp
halt
hamdist
HAngle
harmonic_conjugate
harmonic_division
has
hasattr (builtins)
hash (builtins)
HComplex
HDigits
head
heap_lock (micropython)
heap_unlock (micropython)
Heaviside
Help
help (builtins)
hermite
hessenberg
hessian
hex
hex (builtins)
hexagon
HFormat
hilbert
histogram (matplotlib)
Hmax
Hmin
HMS→
homothety
horner
hprime
HSeparator
HVars
Hwidth
hyp2exp
hyperbola
Hyperbolic
HypT1mean
HypT2mean
HypZ1mean
HypZ1prop
HypZ2mean
HypZ2prop

```

I

```

I
I/O
iabcuv
ibasis
ibpdv
ibpu
ichinrem
ICON
icontent

```

```

id
id (builtins)
IDENMAT
identity
identity (linalg)
idivis
iegcd
iegcd (arit)
if (if block)
IF THEN ELSE END
ifactor
ifactor (arit)
ifactors
IFERR
ifft
ifft (linalg)
IFTE
igcd
ihermite
ilaplace
IM
imag (linalg)
image
ImageDisplay
ImageName
ImageOpacity
ImageXmax
ImageXmin
ImageYmax
ImageYmin
implementation (sys)
implicit_diff
import
in
incircle
Indent
index
index (builtins.bytes)
index (builtins.list)
index (builtins.str)
index (builtins.tuple)
Info
InfType
INPUT
input (builtins)
INSERT
insert
insert (builtins.list)
INSTRING
Instruction
int
int class (builtins)
IntConvCPYR
IntConvEff
IntConvNom
IntCPYR
Integer
Integral
inter
Inter
INTERSECT
intersection (builtins.frozens)
intersection (builtins.set)
intersection_update (builtins.set)
interval2center
inv
inv (linalg)
Inverse
inversion
INVERT
invert (hpprime)
invert_c (hpprime)
INVERT_P
InvestInt
invlaplace
invztrans
IP
iPart
IPYR
iquo
iquorem
irem
IRR
is
is_collinear

```



```

is_concyclic
is_conjugate
is_coplanar
is_cycle
is_element
is_equilateral
is_harmonic
is_harmonic_circle_bundle
is_harmonic_line_bundle
is_isosceles
is_orthogonal
is_parallel
is_parallelogram
is_permu
is_perpendicular
is_rectangle
is_rhombus
is_square
isalpha (builtins.bytes)
isalpha (builtins.str)
ISCHECK
isdigit (builtins.bytes)
isdigit (builtins.str)
isdisjoint (builtins.frozens)
isdisjoint (builtins.set)
Isect
ISECT
isenabled (gc)
isfinite
isfinite (math)
isinf
isinf (math)
isinstance (builtins)
ISKEYDOWN
islower (builtins.bytes)
islower (builtins.str)
ismith
isnan
isnan (math)
isobarycenter
isopolygon
isosceles_triangle
isprime
isprime (arit)
isspace (builtins.bytes)
isspace (builtins.str)
issubclass (builtins)
issubset (builtins.frozens)
issubset (builtins.set)
issuperset (builtins.frozens)
issuperset (builtins.set)
isupper (builtins.bytes)
isupper (builtins.str)
items (builtins.dict)
items (ucollections.OrderedDict)
iter (builtins)
ITERATE
ithprime

```

J

```

J
jacobi_symbol
join
join (builtins.bytes)
join (builtins.str)
jordan
JordanBlock

```

K

```

K
kbd_intr (micropython)
ker
KEY
Keyboard
keyboard (hprime)
keys (builtins.dict)
keys (ucollections.OrderedDict)
KILL
kolmogorovd
kolmogorovt

```

L

L
l1norm
l2norm
Labels
lagrange
laguerre
lambda
Language
laplace
laplacian
latex
lcm
lcm (arit)
lcoeff
ldexp
ldexp (math)
LEFT
left
legendre
legendre_symbol
len (builtins)
length
Length
lgamma (math)
lgcd
LifeAsset
Light
limit
Limits
lin
linalg
Line
line
LINE
line (hprime)
line_c (hprime)
LINE_P
linear_interpolate
linear_regression
linear_regression_plot (matplotlib)
LinearSlope
LinearYIntercept
LineHorz
LineTan
LineVert
LinRegrTConfInt
LinRegrTConfSlope
LinRegrTMeanResp
LinRegrTPredInt
LinRegrTTest
LinSolve
linsolve
linspace (linalg)
List
list class (builtins)
list2mat
ListToMat
l1l_reduce
LN
lname
lncollect
lnexpand
LNP1
LOCAL
locals (builtins)
locus
LOG
log (cmath)
log (math)
log10
log10 (cmath)
log10 (math)
log2 (math)
logarithmic_regression
logb
logistic_regression
Loop
LOWER
lower (builtins.bytes)
lower (builtins.str)
LQ
LSolution

LSQ
lstrip (builtins.bytes)
lstrip (builtins.str)
LSystem
LU
lu
lvar

M

M
Macaulay
MAKELIST
MAKEMAT
MANT
map
Margin
Markup
Mass
mat2list
match (ure)
MatDate
math
Math
matplotl
matpow
matrix
Matrix
matrix (linalg)
MatToList
MAX
max (builtins)
maxnorm
MAXREAL
maxsize (sys)
MaxVal
mean
MeanX
MeanY
Mean₁
Mean₂
Measure
median
median_line
MedVal
mem_alloc (gc)
mem_free (gc)
mem_info (micropython)
member
MEMORY
memoryview class (builtins)
Menu
Messaging
Method
micropython
MicroPython
MID
midpoint
MIN
min (builtins)
MINREAL
MINUS
MinVal
MIRR
mkisom
MKSA
MOD
modf (math)
modgcd
Modified
modules (sys)
More
MOUSE
mouse (hprime)
mRow
MS
MSerr
MSGBOX
mul (linalg)
mult_c_conjugate
mult_conjugate
multinomial

N

N
namedtuple class (ucollections)
NbItem
NbPmt
nDeriv
negbinomial
negbinomial_cdf
negbinomial_icdf
newton
NewValue
next (builtins)
nextprime
nextprime (arit)
NFV
Nmax
Nmin
NomInt
nonlocal
nop
normal
NORMALD
NORMALD_CDF
NORMALD_ICDF
normalize
not
NOT
Notes
NotImplemented
nprimes (arit)
NPV
NTHROOT
Num
Numbers
numer
Numerals
NumIndep
NumStart
NumStep
NumType
NumXStart
NumXStep
NumXZoom
NumYStart
NumYStep
NumYZoom
NumZoom
NUS
n₁
n₂

O

O
object class (builtins)
ObsList
ObsMat
oct (builtins)
odd
odesolve
OldValue
On
ones (linalg)
op
open (builtins)
open (uio)
open_polygon
Operators
opt_level (micropython)
or
OR
ord
ord (builtins)
order_size
OrderedDict class (ucollections)
ordinate
orthocenter
orthogonal
Other

P

P
ploc2
plop2
pa2b2
pack (ustruct)
pack_into (ustruct)
pade
parabola
parallel
parallelogram
parameq
part
partfrac
partition (builtins.bytes)
partition (builtins.str)
pass
Paste
path (sys)
Payback
pcoeff
pcoeff (linalg)
peektime (utimeq)
PercentChange
PercentMargin
PercentMarkup
PercentTotal
perimeter
perimeterat
PERM
perminv
permu2cycles
permu2mat
permuorder
perpen_bisector
perpendicular
peval (linalg)
phase (cmath)
Physics
pi (cmath)
pi (linalg)
pi (math)
PIECEWISE
pivot
Pixels
PIXOFF
PIXOFF_P
PIXON
pixon (hpprime)
pixon_c (hpprime)
PIXON_P
platform (sys)
Plot
plot (matplotlib)
plotcontour
plotfield
plotfunc
plotimplicit
plotinequation
plotlist
PlotMethod
plotode
plotparam
plotpolar
plotseq
pmin
PMT
Point
point
point2d
POISSON
POISSON_CDF
POISSON_ICDF
polar
polar (cmath)
polar_coordinates
polar_point
pole
poly2symb
POLYCOEF
POLYEVAL
polyEval
POLYFORM

Polygon
 polygon
 polygonplot
 polygonsscatterplot
 Polynomial
 polynomial_regression
 POLYROOT
 Pooled
 pop
 pop (builtins.dict)
 pop (builtins.list)
 pop (builtins.set)
 pop (ucollections.OrderedDict)
 pop (utimeq)
 popitem (builtins.dict)
 popitem (ucollections.OrderedDict)
 popleft (ucollections)
 POS
 potential
 pow (builtins)
 pow (math)
 pow2exp
 Power
 power_regression
 powerpc
 powexpand
 powmod
 PPYR
 PredX
 PredY
 Prefix
 prepend
 Pressure
 preval
 prevprime
 prevprime (arit)
 Price
 PriceBond
 Prime
 primpart
 PRINT
 print (builtins)
 PRINT2D
 print_exception (sys)
 Prob
 Probability
 ProbList
 product
 Profit
 Program Editor
 Programs
 projection
 proot
 proot (linalg)
 property class (builtins)
 propfrac
 Psi
 ptayl
 purge
 push (utimeq)
 PUT
 PV
 PX-C
 pystack_use (micropython)
 PYTHON
 python_compat

Q

Q
 q2a
 QPI
 QR
 qstr_info (micropython)
 QuadDelta
 quadrilateral
 QuadSolve
 quantile
 Quantity
 Quantum
 quartile1
 quartile3
 quartiles
 quo

quorem
QUOTE
quote
 Q_1
 Q_3

R

R
radians (math)
Radiation
radical_axis
radius
raise (statement)
rand
rand (linalg)
randbinomial
randchisquare
randexp
randfisher
randgeometric
RANDINT
randint (urandom)
RANDMAT
randnorm
RANDNORM
RANDOM
Random
random (urandom)
randperm
randpoisson
randpoly
randrange
randrange (urandom)
RANDSEED
randstudent
randvector
range
range class (builtins)
RANK
ranm
ratnormal
RE
read (uio.BytesIO)
read (uio.FileIO)
read (uio.StringIO)
read (uio.TextIOWrapper)
readinto (uio.BytesIO)
readinto (uio.FileIO)
readinto (uio.StringIO)
readinto (uio.TextIOWrapper)
READLINE
readline (uio.BytesIO)
readline (uio.FileIO)
readline (uio.StringIO)
readline (uio.TextIOWrapper)
readlines (uio.FileIO)
readlines (uio.TextIOWrapper)
real (linalg)
Recenter
reciprocation
RECT
rect (cmath)
rect (hprime)
rect_c (hprime)
RECT_P
rectangle
rectangular_coordinates
REDIM
reduced_conic
ref
reflection
regroup
REGRS
rem
remove
remove (builtins.list)
remove (builtins.set)
reorder
REPEAT
REPLACE
replace (builtins.bytes)
replace (builtins.str)
repr (builtins)

```

Resid
residue
Resource
restart
Restore
Result
resultant
RETURN
return (statement)
REVERSE
reverse (builtins.list)
reversed (builtins)
revlist
Rewrite
rfind (builtins.bytes)
rfind (builtins.str)
RGB
rhombus
RIGHT
right
right_triangle
rindex (builtins.bytes)
rindex (builtins.str)
RiskFree
romberg
Root
ROOT
ROTATE
rotation
ROUND
round (builtins)
Row
row
rowAdd
rowDim
RowHeight
ROWNORM
rownorm
rowSpace
rowSwap
rpartition (builtins.bytes)
rpartition (builtins.str)
RREF
rref
rref (linalg)
rsolve
rsplit (builtins.bytes)
rsplit (builtins.str)
rstrip (builtins.bytes)
rstrip (builtins.str)
R→B

```

S

```

S
SafeInt
SalePrice
SalvageAsset
SAS
SCALE
SCALEADD
scatterplot (matplotlib)
SCHUR
sCov
ScrollText
search (ure)
SEC
seed (urandom)
seek (uio.BytesIO)
seek (uio.FileIO)
seek (uio.StringIO)
seek (uio.TextIOWrapper)
segment
select
SemiAnnual
seq
seqsolve
SERIAL
series
serrInter
serrLine
serrSlope
serrX
serrY

```



```

set
set class (builtins)
set_cartesian (hprime)
setattr (builtins)
SETBASE
SETBITS
SetDate
setdefault (builtins.dict)
setdefault (ucollections.OrderedDict)
SetDepend
SetFreq
SetIndep
SetSample
setter (builtins.property)
sha256 class (uhashlib)
shape (linalg)
shift
Shift
shift_phase
show (matplotlib)
show_screen (graphic)
shuffle
Si
SideA
SideB
SideC
SIGN
signature
Signed
SignedArea
similarity
simplex_reduce
simplify
simult
SIN
sin (cmath)
sin (math)
sin2costan
sincos
single_inter
SINH
sinh (math)
SIZE
size (linalg)
slice class (builtins)
Slope
SLOPE
slope
slopeat
SOLVE
solve
Solve
solve (linalg)
Solve2x2
Solve3x3
SolveForSlope
SolveForYIntercept
SORT
sort
sort (builtins.list)
sorted (builtins)
Special
SPECNORM
SPECRAD
Speed
spline
split
split (builtins.bytes)
split (builtins.str)
splitlines (builtins.bytes)
splitlines (builtins.str)
sq
sqrfree
SQRT
sqrt
sqrt (cmath)
sqrt (math)
square
srand
SS
SSA
SSerr
SSS
ssX
ssY

```

```

stack_use (micropython)
STARTAPP
startswith (builtins.bytes)
startswith (builtins.str)
STARTVIEW
STAT1
STAT2
staticmethod (builtins)
stddev
stddevp
stderr (sys)
stdin (sys)
stdout (sys)
sto
StockPrice
str class (builtins)
strblit (hprime)
strblit2 (hprime)
strblit2_c (hprime)
strblit_c (hprime)
StrikePrice
STRING
String
STRINGFROMID
StringIO class (uio)
strip
strip (builtins.bytes)
strip (builtins.str)
STUDENT
STUDENT_CDF
STUDENT_ICDF
sturm
sturmab
sturmseq
SUB
sub (linalg)
SUBGROB
SUBGROB_P
subMat
subst
SUM
sum
sum (builtins)
sum_riemann
super (builtins)
SUPPRESS
suppress
surd
SVD
SVL
SWAPCOL
SWAPROW
sX
sY
sylvester
Symb
symb2poly
symmetric_difference (builtins.frozens)
symmetric_difference (builtins.set)
symmetric_difference_update (builtins.set)
SyntaxError class
sys
S1
S2

```

T

```

T
table
tabvar
tail
TAN
tan (math)
tan2cossin2
tan2sincos
tan2sincos2
tangent
TANH
tanh (math)
taylor
tchebyshev1
tchebyshev2
tcoeff
tcollect

```

```

TDim
tell (uio.FileIO)
tell (uio.TextIOWrapper)
Temperature
Template
Terminal
Test
TestScore
TestValue
TEVAL
texexpand
text (matplotlib)
TextIOWrapper class (uio)
TEXTOUT
textout (hprime)
textout_c (hprime)
TEXTOUT_P
TEXTSIZE
Thanks
Theme
threshold (gc)
TICKS
ticks (hprime)
Time
time
TimeMarket
tlin
Tmax
Tmin
to_bytes (builtins.int)
TOff
Torque
Total
TotalCF
Trace
TRACE
Transform
translation
transpose
transpose (linalg)
triangle
TRIANGLE
TRIANGLE_P
trig2exp
trigcos
trigexpand
Trigonometry
trigsin
trigtan
TriType
TRN
trunc
trunc (math)
TRUNCATE
truncate
try (try-except-else-finally block)
tsimplify
Tstep
tuple class (builtins)
TvmFV
TvmIPYR
TvmNbPmt
TvmPMT
TvmPV
TYPE
type
type class (builtins)

```

U

```

U
ucollections
uerrno
UFACTOR
uhashlib
uio
unapply
UNCHECK
uniform
uniform (urandom)
uniform_cdf
uniform_icdf
UNION
union (builtins.frozens)

```

```

union (builtins.set)
Units
unpack (ustruct)
unpack_from (ustruct)
UNTIL
UPART
update (builtins.dict)
update (builtins.set)
update (ucollections.OrderedDict)
update (uhashlib.sha256)
UPPER
upper (builtins.bytes)
upper (builtins.str)
urandom
ure
USB0pen
USBRecv
USBSend
User
USIMPLIFY
ustruct
utimeq
utimeq class (utimeq)
UVAL

```

V

```

V
valuation
values (builtins.dict)
values (ucollections.OrderedDict)
vandermonde
Variable
VariableCost
Variables
variance
Vars
vector
Vector
vector (matplotlib)
VERSION
version (sys)
version_info (sys)
vertices
vertices_abca
View
VIEW
Viscosity
Volatility
Volume
vpotential

```

W

```

W
WAIT
weibull
weibull_cdf
weibull_icdf
when
WHILE
while (loop)
wilcoxonp
wilcoxons
wilcoxont
with
write (uio.BytesIO)
write (uio.FileIO)
write (uio.StringIO)
write (uio.TextIOWrapper)

```

X

```

X
X,T,θ,N
x^y
Xlist
Xmax
Xmin
XOR
XPON
Xtick
Xval

```

Xzoom
 x^2
 x_1
 x_2
 $x_{\square}$

Y

Y
yield
YieldBond
Ylist
Ymax
Ymin
Ytick
Yval
Yzoom

Z

Z
zeros
zeros (linalg)
Zeta
zip
zip (builtins)
Zoom
ztrans

Δ

Δ LIST

Θ

$\theta_{\max}$
 $\theta_{\min}$
 θ_{step}

M

μ_0

Π

π
 Π LIST
 π_0

Σ

Σ
 σ_{Cov}
 Σ LIST
 ΣX
 σX
 ΣX^2
 ΣXY
 ΣY
 σY
 ΣY^2
 σ_1
 σ_2

#

!
!, ∞ , $\rightarrow$
!=
%
%-formatting operator
%=
%CHANGE
%TOTAL
&
&=
' $\square$ '
*
**
**=

$\ast =$
 $+$
 $+/-$
 $+=$
 $,$
 $-$
 $- =$
 $.$
 $. \ast$
 $. +$
 $. -$
 $. /$
 $. ^$
 $/$
 $//$
 $// =$
 $/ =$
 $:=$
 $<$
 $<<$
 $<< =$
 $< =$
 $=$
 $==$
 $>$
 $> =$
 $>>$
 $>> =$
 $^$
 $^ =$
 $|$
 $| =$
 $\sim$
 $\div$
 $\rightarrow$ HMS
 ∂
 ∞
 $\int$
 $\approx$
 $\neq$
 $\leq$
 $\leq, \geq, \neq$
 $\sqsubset$
 $\blacktriangleright$
 $\square$

Créditos

Edición preparada para la comunidad HP Prime a partir de los recursos de ayuda integrados proporcionados para análisis. Su objetivo es facilitar consulta, estudio, programación y preservación técnica de la documentación accesible desde la calculadora.