

IBM Macros para IBM EMM  
Versión 9 Release 0  
11 de diciembre de 2012

*Guía del usuario*



**Note**

Before using this information and the product it supports, read the information in "Notices" en la página 257.

Esta edición se aplica a la versión 9, release 0, modificación 0 de la familia de productos IBM Campaign y a todos los releases y modificaciones subsiguientes hasta que se indique lo contrario en nuevas ediciones.

© Copyright IBM Corporation 1998, 2012.

# Contenido

## Capítulo 1. Utilización de macros en IBM

<b>Campaign</b> . . . . .	<b>1</b>
Resúmenes de función de macro. . . . .	1
Funciones estadísticas . . . . .	1
Funciones matemáticas y trigonométricas. . . . .	2
Funciones de cadena. . . . .	6
Funciones de fecha y hora . . . . .	7
Agrupación de funciones . . . . .	8
Funciones varias . . . . .	8
Parámetros de función de macro. . . . .	9
Especificaciones de formato . . . . .	9
Utilización de las constantes . . . . .	9

## Capítulo 2. Utilización de macros en IBM Interact

<b>IBM Interact</b> . . . . .	<b>11</b>
Resúmenes de función de macro . . . . .	11
Funciones estadísticas . . . . .	11
Funciones matemáticas y trigonométricas . . . . .	12
Funciones de cadena . . . . .	13
Funciones de fecha y hora . . . . .	14
Funciones varias. . . . .	14
Parámetros de función de macro . . . . .	15
Especificaciones de formato . . . . .	15
Utilización de las constantes. . . . .	15

## Capítulo 3. Utilización de macros en IBM PredictiveInsight

<b>IBM PredictiveInsight</b> . . . . .	<b>17</b>
Resúmenes de función de macro . . . . .	17
Funciones estadísticas . . . . .	17
Funciones matemáticas y trigonométricas . . . . .	19
Funciones de ingeniería . . . . .	22
Funciones de cadena . . . . .	23
Funciones de fecha y hora . . . . .	24
Agrupación de funciones. . . . .	25
Funciones varias. . . . .	25
Notas sobre páginas de referencia de macro . . . . .	27
Parámetros de función de macro . . . . .	27
Especificaciones de formato . . . . .	27
Utilización de los rangos de celdas y de columnas . . . . .	28
Utilización de las constantes. . . . .	29
Precisión computacional . . . . .	30
Resultados de celda no válidos y celdas en blanco . . . . .	30

## Capítulo 4. IBM Referencia de macros

<b>33</b>	<b>33</b>
ABS . . . . .	33
ACOS . . . . .	34
ACOT . . . . .	36
ADD_MONTHS . . . . .	37
ALIGN . . . . .	38
AND . . . . .	39
ASIN . . . . .	40
ATAN . . . . .	42

AVG. . . . .	44
AVG_DEV. . . . .	46
BETWEEN. . . . .	48
BIT_AND . . . . .	49
BIT_NOT . . . . .	50
BIT_OR. . . . .	51
BIT_XOR . . . . .	53
BUFFER . . . . .	55
CEILING . . . . .	56
COLUMNA . . . . .	57
CONSTANTE. . . . .	59
COS . . . . .	60
COSH . . . . .	62
COT. . . . .	64
RECUENTO . . . . .	65
RECUENTO_DIFF . . . . .	66
COV. . . . .	68
CURRENT_DATE . . . . .	69
CURRENT_DAY. . . . .	70
CURRENT_JULIAN . . . . .	70
MES_ACTUAL . . . . .	71
TIEMPO_ACTUAL . . . . .	72
Configuración de la fecha en su aplicación web . . . . .	72
CURRENT_WEEKDAY . . . . .	73
CURRENT_YEAR . . . . .	74
CV_FOLDS . . . . .	74
DATALINK . . . . .	76
DATE . . . . .	77
DATE_FORMAT. . . . .	80
DATE_JULIAN . . . . .	81
DATE_STRING . . . . .	82
DAY_BETWEEN. . . . .	83
DAY_FROMNOW . . . . .	84
DAY_INTERVAL . . . . .	85
DAYOF. . . . .	86
DDELINK . . . . .	86
DECIMATE . . . . .	87
DELAY . . . . .	88
DERIVATIVE . . . . .	90
DISTANCE . . . . .	91
DISTINCT. . . . .	92
DIV . . . . .	92
EQ . . . . .	94
EXP . . . . .	96
EXTERNALCALLOUT . . . . .	97
EXTRACT . . . . .	98
FACTORIAL. . . . .	100
FLOOR . . . . .	101
FORMAT. . . . .	102
FRACTION . . . . .	105
GAUSS . . . . .	106
GAUSS_AREA . . . . .	108
GE . . . . .	111
GRID . . . . .	112
GROUPBY . . . . .	114
GROUPBY_WHERE . . . . .	116

GT . . . . .	117	RANK . . . . .	198
HISTOGRAM . . . . .	119	REPEAT . . . . .	200
IF . . . . .	121	ROTATE_LEFT . . . . .	202
IN . . . . .	122	ROTATE_RIGHT . . . . .	203
INIT . . . . .	123	ROUND . . . . .	204
INT . . . . .	125	ROWNUM . . . . .	205
INTEGRAL . . . . .	126	RTRIM . . . . .	206
INVERSE . . . . .	127	SAMPLE_RANDOM . . . . .	206
IS . . . . .	128	SELECT . . . . .	207
ISERROR . . . . .	129	SIGN . . . . .	209
ISEVEN . . . . .	130	SIN . . . . .	210
ISMEMBER . . . . .	131	SINH . . . . .	212
ISODD . . . . .	133	DESVÍO . . . . .	213
CURTOSIS . . . . .	134	SLIDE_WINDOW . . . . .	215
LAG . . . . .	136	SORT . . . . .	217
LE . . . . .	138	SQRT . . . . .	219
LIKE . . . . .	139	STAT . . . . .	220
LN o LOG . . . . .	141	STDV o STDEV . . . . .	223
LOG2 . . . . .	143	STRING_CONCAT . . . . .	225
LOG10 . . . . .	144	STRING_HEAD . . . . .	226
LOWER . . . . .	145	STRING_LENGTH . . . . .	227
LT . . . . .	146	PROPIEDAD_DE_SERIE . . . . .	229
LTRIM . . . . .	147	STRING_SEG . . . . .	229
MAX . . . . .	148	STRING_TAIL . . . . .	230
MAXINDEX . . . . .	150	SUBSAMPLE . . . . .	232
MEAN . . . . .	151	SUBSTITUTE (SUSTITUIR) . . . . .	233
MERGE . . . . .	153	SUBSTR o SUBSTRING . . . . .	234
MIN . . . . .	155	SUM . . . . .	235
MINUS . . . . .	156	TAN . . . . .	237
MOD . . . . .	158	TANH . . . . .	238
MONTHOF . . . . .	160	TO (A) . . . . .	240
MULT . . . . .	160	TOTAL . . . . .	241
NE . . . . .	162	TRANSPOSE (TRANSPONER) . . . . .	243
NORM_MINMAX . . . . .	164	TRUNCATE (TRUNCAR) . . . . .	244
NORM_SIGMOID . . . . .	167	UPPER . . . . .	245
NORM_ZSCORE . . . . .	171	VARIANZA . . . . .	246
NOT . . . . .	174	WEEKDAY (DÍASEM) . . . . .	248
NPV . . . . .	175	WEEKDAYOF . . . . .	249
NUMBER . . . . .	177	XOR . . . . .	250
OFFSET . . . . .	185	XTAB . . . . .	252
OR . . . . .	186	YEAROF . . . . .	253
PCA . . . . .	188		
PCA_FEATURES . . . . .	189	<b>Contacting IBM technical support . . . . .</b>	<b>255</b>
POSITION . . . . .	191		
PLUS . . . . .	193	<b>Notices . . . . .</b>	<b>257</b>
POW . . . . .	194	Trademarks . . . . .	259
RANDOM . . . . .	196	Privacy Policy and Terms of Use Considerations . . . . .	259
RANDOM_GAUSS . . . . .	197		

---



## Capítulo 1. Utilización de macros en IBM Campaign

Este capítulo proporciona información de uso acerca de los macros disponibles para utilizar en IBM® Campaign. Todos los usuarios IBM Campaign deben leer este capítulo antes de intentar utilizar el resto de esta guía.

---

### Resúmenes de función de macro

En las tablas de esta sección se resumen las funciones de macro por categoría.

**Importante:** No todas las macros en esta guía están disponibles en IBM Campaign. Las macros que sólo están disponibles en IBM PredictiveInsight están señaladas mediante este icono:  Las macros que sólo están disponibles en IBM Interact están señaladas mediante este icono: 

Se proporcionan páginas de referencia detalladas para cada función de macro en orden alfabético, comenzando por Capítulo 4, “IBM Referencia de macros”, en la página 33. “Parámetros de función de macro” en la página 27 proporciona información sobre los parámetros de entrada de función de macro.

#### Información relacionada:

“Funciones estadísticas”

“Funciones matemáticas y trigonométricas” en la página 2

“Funciones de cadena” en la página 6

“Funciones de fecha y hora” en la página 7

“Agrupación de funciones” en la página 8

“Funciones varias” en la página 8

### Funciones estadísticas

Nombre de la macro	Devuelve	Descripción
AVG	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula la media aritmética o el promedio de un rango de celdas
COUNT	Valor único en una nueva columna.	Cuenta el número de valores en un rango de datos especificado

Nombre de la macro	Devuelve	Descripción
MAX	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula el número máximo de un rango de celdas
MEAN	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula la media aritmética o el promedio de un rango de celdas
MIN	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula el número mínimo de un rango de celdas
STDV o STDEV	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con un valor para cada fila para la palabra clave ROW.	Calcula la desviación estándar de un rango de celdas
VARIANCE	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula la varianza de un rango de celdas

## Funciones matemáticas y trigonométricas

Nombre de la macro	Devuelve	Descripción
ABS	Una columna para cada columna de entrada	Calcula el número NOT lógico del contenido del rango de datos especificado valor absoluto del contenido del rango de datos especificado

Nombre de la macro	Devuelve	Descripción
ACOS	Una columna para cada columna de entrada	Calcula el arcocoseno del contenido en el rango de datos especificado
ACOT	Una columna para cada columna de entrada	Calcula la arcotangente del contenido del rango de datos especificado
ASIN	Una columna para cada columna de entrada	Calcula el arcoseno del contenido del rango de datos especificado
ATAN	Una columna para cada columna de entrada	Calcula la arcotangente del contenido del rango de datos especificado
AVG	Una columna para cada columna de entrada	Calcula la media aritmética o el promedio de las celdas del rango de datos especificado
BETWEEN	Una columna para cada columna de entrada	Compara dos valores para determinar si el valor proporcionado se encuentra entre otros dos valores
CEILING	Una columna para cada columna de entrada	Calcula el ceiling de cada valor del rango de datos especificado
COLUMN	Una columna para cada columna de entrada	Crea columnas nuevas, concatenando verticalmente los valores de entrada en cada columna
COS	Una columna para cada columna de entrada	Calcula el coseno del contenido del rango de datos especificado
COSH	Una columna para cada columna de entrada	Calcula el número coseno hiperbólico del contenido del rango de datos especificado
COT	Una columna para cada columna de entrada	Calcula la cotangente del contenido del rango de datos especificado
COUNT	Una columna que contiene un solo valor	Cuenta la cantidad de celdas que contienen valores en el rango de datos especificado
EXP	Una columna para cada columna de entrada	Calcula el número natural (e) elevado al contenido de cada celda del rango de datos especificado
FACTORIAL	Una columna para cada columna de entrada	Calcula el factorial de cada valor del rango de datos especificado
FLOOR	Una columna para cada columna de entrada	Calcula el floor de cada valor del rango de datos especificado
FRACTION	Una columna para cada columna de entrada	Devuelve la parte fraccionaria de cada valor del rango de datos especificado

Nombre de la macro	Devuelve	Descripción
INT	Una columna para cada columna de entrada	Calcula el valor entero (redondeado a la baja) del contenido del rango de datos especificado
INVERSE	Una columna para cada columna de entrada	Calcula el negativo del contenido del rango de datos especificado
LN	Una columna para cada columna de entrada	Calcula el logaritmo natural del contenido del rango de datos especificado
LOG	Una columna para cada columna de entrada	Calcula el logaritmo natural del contenido del rango de datos especificado
LOG2	Una columna para cada columna de entrada	Calcula el logaritmo en base 2 del contenido del rango de datos especificado
LOG10	Una columna para cada columna de entrada	Calcula el logaritmo en base 10 del contenido del rango de datos especificado
MAX	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula el número máximo de un rango de celdas
MEAN	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula la media aritmética o el promedio de un rango de celdas
MIN	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula el número mínimo de un rango de celdas
RANDOM	Una columna con el número especificado de valores	Devuelve el número especificado de valores aleatorios
RANDOM_GAUSS	Una columna con el número especificado de valores	Devuelve el número especificado de valores aleatorios de una distribución gaussiana



Nombre de la macro	Devuelve	Descripción
ROUND	Una columna para cada columna de entrada	Calcula el valor redondeado del contenido del rango de datos especificado
SIGN	Una columna para cada columna de entrada	Calcula el signo (positivo o negativo) de los valores del rango de datos especificado
SIN	Una columna para cada columna de entrada	Calcula el seno del contenido del rango de datos especificado
SINH	Una columna para cada columna de entrada	Calcula el seno hiperbólico del contenido del rango de datos especificado
SQRT	Una columna para cada columna de entrada	Calcula la raíz cuadrada del contenido del rango de datos especificado
STDV o STDEV	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula la desviación estándar de un rango de celdas
SUM	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula la suma de un rango de celdas
TAN	Una columna para cada columna de entrada	Calcula la tangente del contenido del rango de datos especificado
TANH	Una columna para cada columna de entrada	Calcula la tangente hiperbólica del contenido del rango de datos especificado
TOTAL	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula la suma de un rango de celdas
TRUNCATE	Una columna para cada columna de entrada	Devuelve la parte no fraccionaria de cada valor del rango de datos especificado

Nombre de la macro	Devuelve	Descripción
VARIANCE	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con un valor para cada fila para la palabra clave ROW.	Calcula la varianza de un rango de celdas

## Funciones de cadena

Nombre de la macro	Devuelve	Descripción
FORMAT	Una columna para cada columna de entrada	Proporciona un control de formato de salida tanto para los números como para las cadenas (tales como las anchura de salida, alineación, precisión numérica, símbolo de separador decimal, separador de miles y así sucesivamente). Devuelve la cadena de salida formateada.
LIKE	Una columna para cada columna de entrada	Determina si una cadena de texto coincide con un patrón especificado
LOWER	Una columna para cada columna de entrada	Convierte el valor de cadena en minúscula
LTRIM	Una columna para cada columna de entrada	Elimina los caracteres de espacio interlineado de cada valor de cadena
NUMBER	Una columna para cada columna de entrada	Convierte cadenas de texto ASCII para horas y fechas en valores numéricos
POSITION	Una columna para cada columna de entrada	Devuelve la posición inicial de un patrón en una cadena de texto
RTRIM	Una columna para cada columna de entrada	Elimina los caracteres de espacio final de cada valor de cadena
STRING_CONCAT	Una columna con un valor para cada fila de la columna de entrada más corta	Concatena cadenas de texto de los rangos de datos especificados
STRING_HEAD	Una columna para cada columna de entrada	Devuelve los primeros <i>n</i> caracteres para cada cadena en el rango de datos especificado
STRING_LENGTH	Una columna para cada columna de entrada	Devuelve la longitud de cada cadena en el rango de datos especificado

Nombre de la macro	Devuelve	Descripción
STRING_PROPER	Una columna para cada columna de entrada	Convierte cada valor de cadena al cambiar la primera letra o cualquier letra que sigue un carácter de espacio en blanco o símbolo (que no sea de subrayado) en mayúscula y todos los otros caracteres en minúsculas
STRING_SEG	Una columna para cada columna de entrada	Devuelve el segmento de la cadena entre dos índices especificados
STRING_TAIL	Una columna para cada columna de entrada	Devuelve los últimos caracteres <i>n</i> de cada cadena en el rango de datos especificado
SUBSTR o SUBSTRING	Una columna para cada columna de entrada	Devuelve los caracteres de una cadena desde una posición inicial
UPPER	Una columna para cada columna de entrada	Convierte el valor de cadena en mayúscula

## Funciones de fecha y hora

Nombre de la macro	Devuelve	Descripción
ADD_MONTHS	Una columna para cada columna de entrada	Devuelve una fecha después de añadir el número especificado de meses
CURRENT_DATE	Una columna para cada columna de entrada	Devuelve la fecha actual en format
CURRENT_DAY	Una columna para cada columna de entrada	Devuelve el día actual del mes como un número del 1 al 31
CURRENT_JULIAN	Una columna para cada columna de entrada	Devuelve el número juliano de la fecha actual
CURRENT_MONTH	Una columna para cada columna de entrada	Devuelve el mes actual del año como un número del 1 al 12
CURRENT_TIME	Una columna para cada columna de entrada	Devuelve la hora actual como una cadena
CURRENT_WEEKDAY	Una columna para cada columna de entrada	Devuelve el día de la semana actual del mes como un número del 0 al 6
CURRENT_YEAR	Una columna para cada columna de entrada	Devuelve el año actual en forma de número
DATE	Una columna para cada columna de entrada	Convierte una cadena de fecha en una fecha juliana
DATE_FORMAT	Una columna para cada columna de entrada	Transforma formatos de fecha
DATE_JULIAN	Una columna para cada columna de entrada	Devuelve la fecha juliana

Nombre de la macro	Devuelve	Descripción
DATE_STRING	Una columna para cada columna de entrada	Devuelve la cadena de fecha de la fecha juliana
DAY_BETWEEN	Una columna para cada columna de entrada	Devuelve el número de días entre dos fechas
DAY_FROMNOW	Una columna para cada columna de entrada	Devuelve el número de días desde la fecha actual a la fecha especificada
DAY_INTERVAL	Una columna para cada columna de entrada	Devuelve el número de días entre dos fechas
DAYOF	Una columna para cada columna de entrada	Devuelve el día del mes en forma de número
MONTHOF	Una columna para cada columna de entrada	Devuelve el mes del año como número
WEEKDAY	Una columna para cada columna de entrada	Convierte cadenas de fecha de texto ASCII en el día de la semana
WEEKDAYOF	Una columna para cada columna de entrada	Devuelve el día de semana de la semana como número
YEAROF	Una columna para cada columna de entrada	Devuelve el año como número

## Agrupación de funciones

Nombre de la macro	Devuelve	Descripción
GROUPBY	Una columna nueva con un valor para cada fila	Resume varias filas de datos dentro de un grupo
GROUPBY_WHERE	Una columna nueva con un valor para cada fila	Resume varias filas de datos que cumplen una condición especificada y que están en un grupo

## Funciones varias

Nombre de la macro	Devuelve	Descripción
IF	Una columna con un valor para cada fila de la columna de entrada más corta	Comienza una sentencia if-then-else condicional
ISERROR	Una columna con un valor para cada fila de la columna de entrada más corta	Devuelve un uno si algún valor en la fila de entrada contiene una celda con error (???). De lo contrario devuelve cero
ISEVEN	Una columna para cada columna de entrada	Comprueba si los valores de entrada son pares (es decir, divisibles por dos)
ISODD	Una columna para cada columna de entrada	Comprueba si los valores de entrada son impares (es decir, no divisibles por dos)

Nombre de la macro	Devuelve	Descripción
ROWNUM	Una columna para cada columna de entrada	Genera números secuenciales desde uno hasta el número de registros

---

## Parámetros de función de macro

Esta sección describe los parámetros y el uso de las funciones de macro en IBM Campaign.

### Especificaciones de formato

Esta sección describe el formato para algunos parámetros comúnmente utilizados. Se aplica a todas las referencias a estos parámetros por especificaciones de funciones de macro en este capítulo.

#### data

El parámetro data representa una columna de datos sobre la que actúa la función de macro. Puede ser una constante o un campo. Consulte la función de macro específica para obtener más detalles.

**Nota:** IBM Campaign no soporta cálculos en varios campos al mismo tiempo o en subconjuntos de filas, como se puede realizar en IBM PredictiveInsight.

Algunos otros nombres de parámetros también utilizan el mismo formato que data. La descripción de estos parámetros hacen referencia a esta sección y formato.

#### palabra clave

El parámetro keyword controla el comportamiento de la función de macro. Indica que puede especificarse una palabra clave (si se omite, se utiliza la predeterminada). Las opciones de palabra clave están listadas para cada función de macro en el siguiente formulario:

**{choice1 | choice2 | choice3}**

Seleccione la opción de palabra clave que proporcione el comportamiento deseado. La opción predeterminada se muestra en negrita. Por ejemplo, dadas las siguientes opciones:

**{RADIANs | DEGREEs}**

Cualquiera de las siguientes funciones de macro son válidas:

COS(V1, RADIANs) COS(V1, DEGREEs)

**Nota:** Muchas funciones de macro toman parámetros de palabra clave {ALL | COL | ROW}. Estas palabras clave no son aplicables en IBM Campaign porque los datos de entrada son siempre una sola columna o campo. La macro siempre se comporta como si se especificara la palabra clave COL. Por lo tanto, no es necesario especificar estas palabras clave al utilizar IBM Campaign.

### Utilización de las constantes

La mayoría de los parámetros de funciones de macro pueden tomar constantes numéricas o expresiones que evalúan una constante numérica (funciones de macro que operan sobre cadenas pueden tomar constantes de cadena).

En las funciones de macro que realizan operaciones de registro por registro (por ejemplo, añadir dos columnas numéricas), la utilización de una constante es equivalente a especificar una columna que contiene ese valor constante en cada fila. Básicamente, cuando se proporciona una constante como un parámetro de entrada, la constante se expande hasta la misma longitud que la entrada.

Algunas funciones de macro pueden tomar cadenas de texto ASCII y constantes numéricas. Los parámetros que pueden aceptar constantes numéricas y cadenas de texto ASCII se indican en la sección "Parámetros" de cada función de macro.

Se proporcionan ejemplos en la siguiente tabla.

Definición de función	Cómo se interpreta la constante
<code>PERCENT_UTILIZ = (CURR_BAL*100)/ CREDIT_LIM</code>	La constante 100 se interpreta como una columna que contiene el mismo número de filas que la columna CURR_BAL, con cada fila que contiene la constante 100. El campo derivado PERCENT_UTILIZ contendrá cada valor de CURR_BAL multiplicado por 100 y dividido entre cada valor de CREDIT_LIM.
<code>NAME = STRING_CONCAT ("Mr. ", LAST_NAME)</code>	La constante "Mr." se interpreta como una columna que contiene el mismo número de filas que la columna LAST_NAME, donde cada fila contiene la constante "Mr.". El campo derivado NAME contendrá cada una de las cadenas de texto en LAST_NAME con el prefijo "Mr.".

---


## Capítulo 2. Utilización de macros en IBM Interact

Esta sección proporciona información de uso acerca de los macros disponibles para utilizar en IBM Interact. Todos los usuarios IBM Interact deben leer esta sección antes de intentar utilizar el resto de esta guía.

---

### Resúmenes de función de macro

En las tablas de esta sección se resumen las funciones de macro en las siguientes categorías.

**Importante:** No todas las macros en esta guía están disponibles en IBM Interact. Las macros que sólo están disponibles en IBM PredictiveInsight están señaladas mediante este icono: 

Se proporcionan páginas de referencia detalladas para cada función de macro en orden alfabético, comenzando por Capítulo 4, “IBM Referencia de macros”, en la página 33. “Parámetros de función de macro” en la página 27 proporciona información sobre los parámetros de entrada de función de macro.

#### Información relacionada:

“Funciones estadísticas”

“Funciones matemáticas y trigonométricas” en la página 12

“Funciones de cadena” en la página 13

“Funciones de fecha y hora” en la página 14

“Funciones varias” en la página 14

### Funciones estadísticas

Nombre de la macro	Devuelve	Descripción
AVG	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con valor para cada fila para la palabra clave ROW.	Calcula la media aritmética o el promedio de un rango de celdas
MAX	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con valor para cada fila para la palabra clave ROW.	Calcula el número máximo de un rango de celdas

Nombre de la macro	Devuelve	Descripción
MEAN	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con valor para cada fila para la palabra clave ROW.	Calcula la media aritmética o el promedio de un rango de celdas
MIN	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con valor para cada fila para la palabra clave ROW.	Calcula el número mínimo de un rango de celdas
STDV o STDEV	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con valor para cada fila para la palabra clave ROW.	Calcula la desviación estándar de un rango de celdas

## Funciones matemáticas y trigonométricas

Nombre de la macro	Devuelve	Descripción
AVG	Una columna para cada columna de entrada	Calcula la media aritmética o el promedio de las celdas del rango de datos especificado
MAX	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula el número máximo de un rango de celdas
MEAN	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula la media aritmética o el promedio de un rango de celdas



Nombre de la macro	Devuelve	Descripción
MIN	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula el número mínimo de un rango de celdas
STDV o STDEV	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula la desviación estándar de un rango de celdas
SUM	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula la suma de un rango de celdas
TOTAL	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula la suma de un rango de celdas

## Funciones de cadena

Nombre de la macro	Devuelve	Descripción
LIKE	Una columna para cada columna de entrada	Determina si una cadena de texto coincide con un patrón especificado
LOWER	Una columna para cada columna de entrada	Convierte el valor de cadena en minúscula
LTRIM	Una columna para cada columna de entrada	Elimina los caracteres de espacio interlineado de cada valor de cadena
NUMBER	Una columna para cada columna de entrada	Convierte cadenas de texto ASCII para horas y fechas en valores numéricos
RTRIM	Una columna para cada columna de entrada	Elimina los caracteres de espacio final de cada valor de cadena

Nombre de la macro	Devuelve	Descripción
STRING_CONCAT	Una columna con un valor para cada fila de la columna de entrada más corta	Concatena las cadenas de los rangos de datos especificados
SUBSTR o SUBSTRING	Una columna para cada columna de entrada	Devuelve los caracteres de una cadena desde una posición inicial
UPPER	Una columna para cada columna de entrada	Convierte el valor de cadena en mayúscula

## Funciones de fecha y hora

Nombre de la macro	Devuelve	Descripción
ADD_MONTHS	Una columna para cada columna de entrada	Devuelve una fecha después de añadir el número especificado de meses
CURRENT_DATE	Una columna para cada columna de entrada	Devuelve la fecha actual en format
CURRENT_DAY	Una columna para cada columna de entrada	Devuelve el día actual del mes como un número del 1 al 31
CURRENT_MONTH	Una columna para cada columna de entrada	Devuelve el mes actual del año como un número del 1 al 12
CURRENT_WEEKDAY	Una columna para cada columna de entrada	Devuelve el día de la semana actual del mes como un número del 0 al 6
CURRENT_YEAR	Una columna para cada columna de entrada	Devuelve el año actual en forma de número
DATE	Una columna para cada columna de entrada	Convierte una cadena de fecha en una fecha juliana
DATE_FORMAT	Una columna para cada columna de entrada	Transforma formatos de fecha

## Funciones varias

Nombre de la macro	Devuelve	Descripción
EXTERNALCALLOUT	Valores según definido por la aplicación personalizada escrita con la API ExternalCallout	Llama a una aplicación personalizada escrita con la API ExternalCallout  Para obtener más información, consulte la Guía del Desarrollador <i>IBM Interact</i> .
IF	Una columna con un valor para cada fila de la columna de entrada más corta	Comienza una sentencia if-then-else condicional

---

## Parámetros de función de macro

Esta sección describe los parámetros y el uso de las funciones de macro en IBM Interact.

### Especificaciones de formato

Esta sección describe el formato para algunos parámetros comúnmente utilizados. Se aplica a todas las referencias a estos parámetros por especificaciones de funciones de macro en esta sección.

#### **data**

El parámetro `data` representa una columna de datos sobre la que actúa la función de macro. Puede ser una constante o un campo. Consulte la función de macro específica para obtener más detalles.

**Nota:** IBM Interact no soporta cálculos en varios campos al mismo tiempo o en subconjuntos de filas, como se puede realizar en IBM PredictiveInsight.

Algunos otros nombres de parámetros también utilizan el mismo formato que `data`. La descripción de estos parámetros hace referencia a esta sección y formato.

#### **palabra clave**

El parámetro `keyword` controla el comportamiento de la función de macro. Indica que puede especificarse una palabra clave (si se omite, se utiliza la predeterminada). Las opciones de palabra clave están listadas para cada función de macro en el siguiente formulario:

```
{choice1 | choice2 | choice3}
```

Seleccione la opción de palabra clave que proporcione el comportamiento deseado. La opción predeterminada se muestra en negrita. Por ejemplo, dadas las siguientes opciones:

```
{RADIANS | DEGREES}
```

Cualquiera de las siguientes funciones de macro son válidas:

```
COS(V1, RADIANS) COS(V1, DEGREES)
```

**Nota:** Muchas funciones de macro toman parámetros de palabra clave {ALL | COL | ROW}. Estas palabras clave no son aplicables en IBM Interact porque los datos de entrada son siempre una sola columna o campo. La macro siempre se comporta como si se especificara la palabra clave COL. Por lo tanto, no es necesario especificar estas palabras clave al utilizar IBM Interact.

### Utilización de las constantes

La mayoría de los parámetros de funciones de macro pueden tomar constantes numéricas o expresiones que evalúan una constante numérica (funciones de macro que operan sobre cadenas pueden tomar constantes de cadena).

En las funciones de macro que realizan operaciones de registro por registro (por ejemplo, añadir dos columnas numéricas), la utilización de una constante es equivalente a especificar una columna que contiene ese valor constante en cada fila. Básicamente, cuando se proporciona una constante como un parámetro de entrada, la constante se expande hasta la misma longitud que la entrada.

Algunas funciones de macro pueden tomar cadenas de texto ASCII así como constantes numéricas. Los parámetros que pueden aceptar constantes numéricas y cadenas de texto ASCII se indican en la sección "Parámetros" de cada función de macro.

Se proporcionan ejemplos en la siguiente tabla.

Definición de función	Cómo se interpreta la constante
PERCENT_UTILIZ = (CURR_BAL*100)/ CREDIT_LIM	La constante 100 se interpreta como una columna que contiene el mismo número de filas que la columna CURR_BAL, con cada fila que contiene la constante 100. El campo derivado PERCENT_UTILIZ contendrá cada valor de CURR_BAL multiplicado por 100 y dividido entre cada valor de CREDIT_LIM.
NAME = STRING_CONCAT ("Mr.", LAST_NAME)	La constante "Mr." se interpreta como una columna que contiene el mismo número de filas que la columna LAST_NAME, donde cada fila contiene la constante "Mr.". El campo derivado NAME contendrá cada una de las cadenas de texto LAST_NAME con prefacio "Mr.".

**Nota:** Las constantes tales como DT\_DELIM\_M\_D\_Y, requieren comillas únicas.

---


## Capítulo 3. Utilización de macros en IBM PredictiveInsight

Este capítulo proporciona información de uso acerca de los macros disponibles para utilizar en IBM PredictiveInsight. Todos los usuarios IBM PredictiveInsight deben leer este capítulo antes de intentar utilizar el resto de la guía.

---

### Resúmenes de función de macro

En las tablas de esta sección se resumen las funciones de macro en las siguientes categorías.

**Importante:** No todas las macros en esta guía están disponibles en IBM PredictiveInsight. Las macros que sólo están disponibles en IBM Interact están señaladas mediante este icono: 

Se proporcionan páginas de referencia detalladas para cada función de macro en orden alfabético, comenzando por Capítulo 4, “IBM Referencia de macros”, en la página 33. “Parámetros de función de macro” en la página 27 proporciona información sobre los parámetros de entrada de función de macro.

#### Información relacionada:

“Funciones estadísticas”

“Funciones matemáticas y trigonométricas” en la página 19

“Funciones de ingeniería” en la página 22

“Funciones de cadena” en la página 23

“Funciones de fecha y hora” en la página 24

“Agrupación de funciones” en la página 8

### Funciones estadísticas

Nombre de la macro	Devuelve	Descripción
AVG	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con un valor para cada fila para la palabra clave ROW.	Calcula la media aritmética o el promedio de un rango de celdas
AVG_DEV	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con un valor para cada fila para la palabra clave ROW.	Calcula la desviación promedio de un rango de celdas

Nombre de la macro	Devuelve	Descripción
HISTOGRAM	Un valor único en una nueva columna	Calcula el histograma de un rango de datos especificado con la utilización de límites de agrupaciones proporcionados
KURTOSIS	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con un valor para cada fila para la palabra clave ROW.	Calcula la curtosis de un rango de celdas
MEAN	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con un valor para cada fila para la palabra clave ROW.	Calcula la media aritmética o el promedio de un rango de celdas
SKEW	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con un valor para cada fila para la palabra clave ROW.	Calcula el desvío de la distribución de un rango de celdas
STAT	Número variable de columnas (ver macro)	Calcula desde el primer al cuarto momento del rango de datos especificado
STDV o STDEV	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con un valor para cada fila para la palabra clave ROW.	Calcula la desviación estándar de un rango de celdas
VARIANCE	Un valor único en una nueva columna para la palabra clave ALL; una columna con un valor único para cada columna de entrada para la palabra clave COL; una columna con un valor para cada fila para la palabra clave ROW.	Calcula la varianza de un rango de celdas

Nombre de la macro	Devuelve	Descripción
XTAB	Una columna para cada valor distinto en el segundo parámetro de entrada con un número de filas igual al número de valores distintos en el primer parámetro de entrada	Calcula valores distintos en dos columnas y devuelve la COUNTOF cada combinación

## Funciones matemáticas y trigonométricas

Nombre de la macro	Devuelve	Descripción
ABS	Una columna para cada columna de entrada	Calcula el número NOT lógico del contenido del rango de datos especificado valor absoluto del contenido del rango de datos especificado
ACOS	Una columna para cada columna de entrada	Calcula el arcocoseno del contenido en el rango de datos especificado
ACOT	Una columna para cada columna de entrada	Calcula la arcotangente del contenido del rango de datos especificado
ASIN	Una columna para cada columna de entrada	Calcula el arcoseno del contenido del rango de datos especificado
ATAN	Una columna para cada columna de entrada	Calcula la arcotangente del contenido del rango de datos especificado
CEILING	Una columna para cada columna de entrada	Calcula el ceiling de cada valor del rango de datos especificado
COS	Una columna para cada columna de entrada	Calcula el número coseno del contenido del rango de datos especificado
COSH	Una columna para cada columna de entrada	Calcula el número coseno hiperbólico del contenido del rango de datos especificado
COT	Una columna para cada columna de entrada	Calcula la cotangente del contenido del rango de datos especificado
COUNT	Una columna que contiene un solo valor	Cuenta la cantidad de celdas que contienen valores en el rango de datos especificado
COV	Un solo valor en una o más columnas	Calcula la covarianza de dos rangos de entrada
DERIVATIVE	Una columna para cada columna de entrada	Calcula la derivada de los valores del rango de datos especificado
DIV	Una columna para cada columna de entrada	Divide un rango de datos especificado por otro

Nombre de la macro	Devuelve	Descripción
EQ	Una columna para cada columna de entrada	Devuelve TRUE si un rango de datos es igual a otro
EXP	Una columna para cada columna de entrada	Calcula el número natural (e) elevado al contenido de cada celda del rango de datos especificado
FACTORIAL	Una columna para cada columna de entrada	Calcula el factorial de cada valor del rango de datos especificado
FLOOR	Una columna para cada columna de entrada	Calcula el floor de cada valor del rango de datos especificado
FRACTION	Una columna para cada columna de entrada	Devuelve la parte fraccionaria de cada valor del rango de datos especificado
GAUSS	Una columna para cada columna de entrada	Calcula la gaussiana de los valores del rango de datos especificado
GAUSS_AREA	Una columna para cada columna de entrada	Calcula el número NOT lógico del contenido del rango de datos especificado área debajo de la gaussiana de los valores del rango de datos especificado
GE	Una columna para cada columna de entrada	Devuelve TRUE si un rango de datos es mayor o igual a otro
GT	Una columna para cada columna de entrada	Devuelve TRUE si un rango de datos es mayor que otro
INT	Una columna para cada columna de entrada	Calcula el valor entero (redondeado a la baja) del contenido del rango de datos especificado
INTEGRAL	Una columna para cada columna de entrada	Calcula la integral de los valores del rango de datos especificado
INVERSE	Una columna para cada columna de entrada	Calcula el negativo del contenido del rango de datos especificado
ISEVEN	Una columna para cada columna de entrada	Comprueba si los valores de entrada son pares (es decir, divisibles por dos)
ISODD	Una columna para cada columna de entrada	Comprueba si los valores de entrada son impares (es decir, no divisibles por dos)
LE	Una columna para cada columna de entrada	Devuelve TRUE si un rango de datos es menor o igual a otro
LN	Una columna para cada columna de entrada	Calcula el logaritmo natural del contenido del rango de datos especificado



Nombre de la macro	Devuelve	Descripción
LOG	Una columna para cada columna de entrada	Calcula el logaritmo natural del contenido del rango de datos especificado
LOG2	Una columna para cada columna de entrada	Calcula el logaritmo en base 2 del contenido del rango de datos especificado
LOG10	Una columna para cada columna de entrada	Calcula el logaritmo en base 10 del contenido del rango de datos especificado
LT	Una columna para cada columna de entrada	Devuelve TRUE si un rango de datos es menor que otro
MAX	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula el número máximo de un rango de celdas
MIN	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula el número mínimo de un rango de celdas
MINUS	Una columna para cada columna de entrada	Resta un rango de datos especificado de otro
MOD	Una columna para cada columna de entrada	Calcula el módulo del contenido del rango de datos especificado
MULT	Una columna para cada columna de entrada	Multiplica el contenido de dos rangos de datos
NE	Una columna para cada columna de entrada	Devuelve TRUE si un rango de datos no es igual a otro
PLUS	Una columna para cada columna de entrada	Suma el contenido de dos rangos de datos
POW	Una columna para cada columna de entrada	Calcula un valor base elevado a las potencias exponenciales especificadas
RANDOM	Una columna con el número especificado de valores	Devuelve el número especificado de valores aleatorios
RANDOM_GAUSS	Una columna con el número especificado de valores	Devuelve el número especificado de valores aleatorios de una distribución gaussiana

Nombre de la macro	Devuelve	Descripción
ROUND	Una columna para cada columna de entrada	Calcula el valor redondeado del contenido del rango de datos especificado
SIGN	Una columna para cada columna de entrada	Calcula el signo (positivo o negativo) de los valores del rango de datos especificado
SIN	Una columna para cada columna de entrada	Calcula el seno del contenido del rango de datos especificado
SINH	Una columna para cada columna de entrada	Calcula el seno hiperbólico del contenido del rango de datos especificado
SQRT	Una columna para cada columna de entrada	Calcula la raíz cuadrada del contenido del rango de datos especificado
SUM	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula la suma de un rango de celdas
TAN	Una columna para cada columna de entrada	Calcula la tangente del contenido del rango de datos especificado
TANH	Una columna para cada columna de entrada	Calcula la tangente hiperbólica del contenido del rango de datos especificado
TOTAL	Valor único en una columna nueva para la palabra clave ALL, una columna con un valor único para cada columna de entrada para la palabra clave COL, una columna con un valor único para cada fila para la palabra clave ROW.	Calcula la suma de un rango de celdas
TRUNCATE	Una columna para cada columna de entrada	Devuelve la parte no fraccionaria de cada valor del rango de datos especificado

## Funciones de ingeniería

Nombre de la macro	Devuelve	Función
DELAY	Una columna para cada columna de entrada	Devuelve los valores de las columnas de entrada retardados por un número especificado de pasos de tiempo

Nombre de la macro	Devuelve	Función
GRID	Una columna para cada columna de entrada	Devuelve una cuadrícula de todas las combinaciones de valores posibles (una por fila)
LAG	Una columna para cada columna de entrada	Devuelve los valores de las columnas de entrada rezagados por un número especificado de pasos de tiempo
NORM_MAXMIN	Una columna para cada columna de entrada	Calcula la normalización min/max de un rango de datos
NORM_SIGMOID	Una columna para cada columna de entrada	Calcula la normalización sigmoïdal de un rango de datos
NORM_ZSCORE	Una columna para cada columna de entrada	Calcula la normalización de la puntuación $z$ de un rango de datos
PCA	Una columna para cada columna de entrada, más una	Calcula los vectores para los componentes principales del rango de datos especificado
PCA_FEATURES	Una columna para cada columna de entrada, más una	Extrae $n$ características del rango de datos especificado
SAMPLE_RANDOM	Una columna para cada columna de entrada	Devuelve columnas de $n$ celdas, cada una de las cuales contiene una muestra aleatoria del rango de datos especificado
SLIDE_WINDOW	Número de columnas de entrada por el parámetro <code>width</code>	Crea un patrón desde una ventana especificada y lo desliza hacia abajo para crear el siguiente patrón
SORT	Una columna para cada columna de entrada	Ordena los valores del rango de datos especificado en orden ascendente o descendente
SUBSAMPLE	Una columna para cada columna de entrada	Reduce los datos al devolver cada valor de fila en la posición $n$

## Funciones de cadena

Nombre de la macro	Devuelve	Función
DISTINCT	Una columna para cada columna de entrada	

Nombre de la macro	Devuelve	Función
FORMAT	Una columna para cada columna de entrada	Proporciona el control del formato de salida tanto para los números como para las cadenas (por ejemplo, anchura de salida, alineación, precisión numérica, símbolo de separador decimal, separador de miles y así sucesivamente). Devuelve la cadena de salida formateada.
LIKE	Una columna para cada columna de entrada	Determina si una cadena de texto coincide con un patrón especificado
NUMBER	Una columna para cada columna de entrada	Convierte las cadenas de texto ASCII para horas y fechas a valores numéricos.
POSITION	Una columna para cada columna de entrada	Devuelve la posición inicial de un patrón en una cadena de texto
STRING_CONCAT	Una columna con un valor para cada fila de la columna de entrada más corta	Concatena cadenas de texto de los rangos de datos especificados
STRING_HEAD	Una columna para cada columna de entrada	Devuelve los primeros <i>n</i> caracteres para cada cadena en el rango de datos especificado
STRING_LENGTH	Una columna para cada columna de entrada	Devuelve la longitud de cada cadena en el rango de datos especificado
STRING_SEG	Una columna para cada columna de entrada	Devuelve el segmento de la cadena entre dos índices especificados
STRING_TAIL	Una columna para cada columna de entrada	Devuelve los últimos caracteres <i>n</i> de cada cadena en el rango de datos especificado
SUBSTR	Una columna para cada columna de entrada	Devuelve los caracteres de una cadena desde una posición inicial
SUBSTRING	Una columna para cada columna de entrada	Devuelve los caracteres de una cadena desde una posición inicial

## Funciones de fecha y hora

Nombre de la macro	Devuelve	Descripción
ADD_MONTHS	Una columna para cada columna de entrada	Devuelve una fecha después de añadir el número especificado de meses
CURRENT_DATE	Una columna para cada columna de entrada	Devuelve la fecha actual en format

Nombre de la macro	Devuelve	Descripción
CURRENT_DAY	Una columna para cada columna de entrada	Devuelve el día actual del mes como un número del 1 al 31
CURRENT_JULIAN	Una columna para cada columna de entrada	Devuelve el número juliano de la fecha actual
CURRENT_MONTH	Una columna para cada columna de entrada	Devuelve el mes actual del año como un número del 1 al 12
CURRENT_TIME	Una columna para cada columna de entrada	Devuelve la hora actual como una cadena
CURRENT_WEEKDAY	Una columna para cada columna de entrada	Devuelve el día de la semana actual del mes como un número del 0 al 6
CURRENT_YEAR	Una columna para cada columna de entrada	Devuelve el año actual en forma de número
DATE	Una columna para cada columna de entrada	Convierte una cadena de fecha en una fecha juliana
DATE_FORMAT	Una columna para cada columna de entrada	Transforma formatos de fecha
MONTHOF	Una columna para cada columna de entrada	Devuelve el mes del año como número
WEEKDAY	Una columna para cada columna de entrada	Convierte cadenas de fecha de texto ASCII en el día de la semana
WEEKDAYOF	Una columna para cada columna de entrada	Devuelve el día de semana de la semana como número
YEAROF	Una columna para cada columna de entrada	Devuelve el año como número

## Agrupación de funciones

Nombre de la macro	Devuelve	Descripción
GROUPBY	Una columna nueva con un valor para cada fila	Resume varias filas de datos dentro de un grupo
GROUPBY_WHERE	Una columna nueva con un valor para cada fila	Resume varias filas de datos que cumplen una condición especificada y que están en un grupo

## Funciones varias

Nombre de la macro	Devuelve	Función
BUFFER	Una columna para cada columna de entrada	Copia el rango de datos de entrada y se actualiza dinámicamente
COLUMN	Una columna para cada columna de entrada	Crea columnas nuevas, concatenando verticalmente los valores de entrada en cada columna

Nombre de la macro	Devuelve	Función
CONSTANT	Una columna para cada columna de entrada	Copia el rango de datos de entrada una vez (sin actualización dinámica)
COUNT_DIFF	Dos columnas	Devuelve cada valor exclusivo en la entrada con un recuento del número de veces que el valor ha aparecido
CV_FOLDS	Una columna con un valor para cada fila de la columna de entrada más corta	Divide los patrones de entrada en $n$ pliegues de datos para su validación cruzada
DATALINK	Una columna para cada columna de datos de enlace	Crea un enlace interno a los datos en una hoja de cálculo IBM PredictiveInsight
DDELINK	Una columna para cada columna de datos de enlace	Crea un enlace externo a los datos de otra aplicación de Windows
DECIMATE	Columnas MAX_VALUE (una columna por cada valor de salida)	Realiza la decimación de una columna de números en varias columnas donde un uno indica el valor de índice.
EXTRACT	Una columna para cada columna de entrada	Extrae filas de acuerdo a los valores de una columna de predicado
IF	Una columna con un valor para cada fila de la columna de entrada más corta	Comienza una sentencia condicional if-then-else de la sentencia
INIT	Una columna	Inicializa valores de incremento temporal anteriores para una función recursiva
ISERROR	Una columna con un valor para cada fila de la columna de entrada más corta	Devuelve un uno si algún valor en la fila de entrada contiene una celda con error (???). De lo contrario devuelve cero
ISMEMBER	Una columna para cada columna de entrada	Comprueba un rango de entrada en una "tabla" de valores, devolviendo un uno si un valor está contenido en la tabla. De lo contrario, devuelve cero.
MAXINDEX	Una columna con un valor para cada fila de la columna de entrada más corta	Devuelve el índice de columna del $n$ -ésimo (primero, segundo, tercero y así sucesivamente) valor máximo para cada fila de la columna especificada

Nombre de la macro	Devuelve	Función
RANK	Una columna para cada columna de entrada	Divide los datos en grupos nbins (el valor predeterminado es 10), cada uno con aproximadamente un número igual de valores distintos y devuelve el grupo en el que cae cada fila de datos
SORT	Una columna para cada columna de entrada	Ordena el rango de datos especificado en orden ascendente o descendente

## Notas sobre páginas de referencia de macro

Cada una de las funciones de macro disponibles se describe en la sección *Resúmenes de funciones de macro* para cada producto. Antes de comenzar a trabajar con una macro, revise las convenciones de sintaxis en el prefacio.

Se proporcionan ejemplos simples para cada función de macro. En estos ejemplos, las columnas adicionales creadas son denominadas VX, VY, VZ, etc. Los nombres reales utilizados en la hoja de cálculo dependen de la situación específica.

---

## Parámetros de función de macro

Esta sección describe los parámetros y el uso de las funciones de macro en IBM PredictiveInsight.

### Especificaciones de formato

Esta sección describe el formato para algunos parámetros comúnmente utilizados. Se aplica a todas las referencias a estos parámetros por especificaciones de funciones de macro en este capítulo.

#### data

El parámetro data representa un rango de datos para que actúe la función de macro. Normalmente puede ser una constante, una columna o un rango de celdas (consulte la función de macro específica para obtener más detalles). El formato del parámetro data es como se describe a continuación:

```
begin_data [: end_data]
```

Donde begin\_data puede ser una constante (por ejemplo, 10.2), el nombre de una columna (por ejemplo, V1) o un rango de celdas (por ejemplo, V1[1:100]). El parámetro end\_data es opcional. Si se lo proporciona, begin\_data se utiliza como un punto de partida y debe ser una columna o un rango de celdas. El punto final es especificado por end\_data.

**Nota:** Algunos otros parámetros también utilizan el mismo formato que data. La definición de estos parámetros hacen referencia a esta sección y formato.

#### palabra clave

El parámetro keyword controla el comportamiento de la función de macro. Indica que puede especificarse una palabra clave (si se omite, se utiliza la predeterminada). Las opciones de palabra clave están listadas para cada función de macro en el siguiente formulario:

{**choice1** | choice2 | choice3}

Seleccione la opción de palabra clave que proporcione el comportamiento deseado. La opción predeterminada se muestra en negrita. Por ejemplo, dado el siguiente formato:

{ **ALL** | COL | ROW }

Cualquiera de las siguientes funciones de macro son válidas:

AVG(V1:V5) **AVG(V1:V5, ALL)** AVG(V1:V5, COL) AVG(V1:V5, ROW)

## Utilización de los rangos de celdas y de columnas

Los rangos de celdas y de columnas pueden proporcionarse como entradas para la mayoría de los parámetros de las funciones de macro. Debe acatar las siguientes reglas:

- El número de columnas debe coincidir
- Los valores de retorno comienzan en la primera celda
- Los rangos de celdas se rellenan automáticamente con ceros

### El número de columnas debe coincidir

Cuando se proporciona dos o más rangos de datos como entrada y se realizan cálculos en columnas, los dos rangos de datos deben contener el mismo número de columnas. De lo contrario, solo se utiliza las dimensiones del menor rango de datos (algunas funciones de macro señalarán un error). Si el rango de datos contiene un número diferente de filas, la mayoría de las funciones de macro realizan cálculos hasta la última fila de la columna más corta inclusive.

- Por ejemplo, con rangos de columnas, la definición de macro V6 = V1:V3 AND V4:V6 genera tres columnas de salida (ambos rangos de datos contienen tres columnas). Se aplica el AND lógico entre la columna V1 y la columna V4, entre la columna V2 y la V5 y entre la columna V3 y la V6. Sin embargo, V6 = V1:V3 AND V4:V5 devuelve solo dos columnas de salida (el primer rango de columnas contiene tres columnas, el segundo contiene solo dos y se utiliza el menor de los dos). En este caso, se aplica el AND lógico entre la columna V1 y la columna V4 y entre la columna V2 y V5. La columna V3 no se utiliza.
- Con rangos de celdas, la definición de macro V7 = V1[1:5]:V2 AND V4[10:50]:V5 genera dos columnas de salida (ambos rangos de entrada tienen dos columnas). Las columnas de salida V7 y V8 contendrán cinco valores (los AND lógicos entre las celdas 1 a 5 y las celdas 10 a 14). La definición de macro V7 = V1[1:5]:V2 AND V4 genera solo una columna de salida, porque el segundo rango de datos solo contiene una columna.

**Importante:** Especificar una columna sin un rango de celdas es equivalente a especificar la columna entera (es decir, un rango de celdas de uno a través de la longitud de la columna).

### Los valores de retorno comienzan en la primera celda

Los valores devueltos por una función de macro se colocan en celdas consecutivas, empezando por la primera celda (por ejemplo, TEMP[1]). Por ejemplo, el cálculo de V2=SIN(V1[100:200]) colocaría 101 valores en las celdas 1-100 de la columna V2.

**Importante:** Si necesita realizar una operación fila por fila en un rango de celdas y desea conservar los resultados en las correspondientes filas (es decir, si opera en celdas [10:20] y desea que los resultados se coloquen en las celdas 10-20 de la



columna resultante), calcule en su lugar específicamente el rango de celdas [1:20]. Así se calcularán algunos valores innecesarios, pero se colocarán los resultados en las filas deseadas.

### Rangos de celdas rellenos con ceros automáticamente

Si especifica un rango de celdas, cualquier blanco (vacío) celdas en el rango de celdas se rellenan automáticamente con ceros. Por ejemplo,  $V3 = V1[1:3] * V2$  produce:

V1 V2 V3 1 2 2 3 4 12 [] 6 0

donde [] representa una celda en blanco (es decir, la columna V1 contiene sólo dos valores de celda). Sin embargo, la especificación de  $V3 = V1 * V2$  podría producir los dos valores 2 y 12 (cálculos se llevará a cabo hasta la más corta de las dos columnas).

## Utilización de las constantes

La mayoría de los parámetros de la función de macro pueden tomar constantes numéricas o expresiones que evalúan una constante numérica (funciones de macro que operan sobre cadenas pueden tomar constantes de cadena). En macro funciones que realizan operaciones registro por registro, la utilización de una constante es equivalente a especificar una columna que contenga ese valor constante en cada fila. Básicamente, cuando se proporcionan como parámetros de entrada una constante y un rango de celdas o de columnas, la constante se expande hasta las mismas dimensiones que el rango de celdas o de columnas. Cualquier columna que contenga una celda única utilizada como entrada en una función de macro, se considera una constante.

Algunas funciones de macro pueden tomar cadenas de texto ASCII así como constantes numéricas. Los parámetros que pueden aceptar constantes numéricas y cadenas de texto ASCII se indican en la sección "Parámetros" de cada función de macro.

Se proporcionan ejemplos en la siguiente tabla.

Definición de función	Cómo se interpreta la constante
$V1=3+5$	Cada una de las constantes se interpreta como una columna única que contiene un valor único. La columna V1 contendrá el valor único 8.
$V2=2 * V1$	La constante 2 se interpreta como una columna que contiene el mismo número de filas que la columna V1, con cada fila que contiene la constante 2. La columna V2 contendrá cada valor de V1 multiplicado por 2.
$V2 = \text{STRING\_CONCAT}(V1, \text{"ing"})$	La constante "ing" se interpreta como una columna que contiene el mismo número de filas que la columna V1, con cada fila que contiene la constante "ing". La columna V2 contendrá cada una de las cadenas de texto en V1 concatenada con "ing".

Definición de función	Cómo se interpreta la constante
V4=V1:V3/ AVG(V1:V3)	La expresión AVG(V1:V3) se evalúa en un valor constante, por ejemplo $x$ . La constante $x$ se interpreta como tres columnas que contienen tantas filas como la columna más corta de V1, V2 o V3. Cada celda contiene la constante $x$ . Las columnas de salida, V4-V6, contendrán los valores de las columnas V1-V3 dividido entre $x$ .
V3=V1[10:20]^2	La constante 2 se interpreta como un rango de celdas con 11 filas, cada una de las cuales contiene el valor 2.

## Precisión computacional

Todos los cálculos en IBM PredictiveInsight hojas de cálculo están limitadas a un máximo de precisión de 32 bits.

### Cálculos con enteros

Las funciones de macro que realizan cálculos con enteros (BIT\_AND, BIT\_NOT, BIT\_OR, BIT\_XOR y T0) no manejan números negativos. Los valores deben estar entre 0 y  $(2^{24} - 1)$ ; de lo contrario, se devuelve un error.

## Resultados de celda no válidos y celdas en blanco

### ¿Las celdas contienen ???

Si cualquier operación de hoja de cálculo produce resultados no válidos, la celda contendrá ??? en lugar de un valor calculado. Por ejemplo, la raíz cuadrada de un número negativo utilizando la función de macro produce SQRT ??? para cada entrada valor negativo.

### ¿Basado en cálculos ?

Después de que una celda contiene valor ???, la mayoría de los cálculos, utilizando dicha celda, propagarán el resultado ???. Por ejemplo, sumar una columna que contiene una o más celdas con ??? produce ???.

Si el resultado ??? aparece en una o más celdas, pulse en una de las celdas que contienen ??? y lea el mensaje de error visualizado en el cuadro de texto **Definición de función**. Consulte la referencia de función de macro en esta guía para ver por qué el error se produce. A continuación, modifique la definición de función para que todos los valores de entrada sean válidos. Por ejemplo, con la raíz cuadrada ejemplo, puede tomar primero el valor absoluto de las entradas:

V2 = SQRT(V1) --> V2 = SQRT(ABS(V1))

**Nota:** Puede que necesite un rastreo inverso de ??? los valores en las columnas dependientes para rectificar el problema.

**Importante:** Cualquier celda que contiene ??? que se pasa al gestor de experimento, como parte de un proceso de capacitación o patrón de prueba, se pasan como ceros.

### Celdas en blanco y celdas ???

Las celdas en blanco son simplemente celdas vacías. Aparecen solo al final de una columna o en las columnas vacías. Las celdas en blanco y las celdas ??? se tratan

de forma diferente por diferentes funciones de macro. La mayoría de las funciones de macro tratan a las celdas en blanco como ceros y señalan y propagan errores en las celdas ??? con las siguientes excepciones:

- La función de macro MAXINDEX ignora las celdas en blanco y las celdas ???.
- La función de macro OFFSET ignora las celdas en blanco y las celdas ???.
- La función de macro DDELINK pasa celdas en blanco y ???.
- Para ver las funciones de macro que omiten celdas en blanco, consulte "Utilización de las constantes" en la página 29. Los errores en las celdas ??? aún se señalan y propagan.
- Para ver las funciones de macro que omiten celdas en blanco, pero utilizan celdas ??? como se ha descrito, consulte "Funciones de macro que se saltan celdas en blanco y usan llamadas "???"".
- Para ver las funciones de macro que devuelven ??? para todas las celdas, si alguna de las celdas de la lista de argumentos contiene ???, consulte "Funciones de macro que no pueden procesar celdas con el símbolo ???" en la página 32.

### Funciones de macro que se saltan celdas en blanco y propagan las celdas con el símbolo ???

AVG	NORM_MINMAX
DERIVATIVE	NORM_ZSCORE
KURTOSIS	PCA
MAX	PCA_FEATURES
MEAN	SKEW
MIN	SUM
MOMENTS	VARIANCE

### Funciones de macro que se saltan celdas en blanco y usan llamadas "???"

Función de macro	Cómo se utilizan las celdas con el símbolo ???.
COUNT	??? Se cuentan las celdas con el símbolo ???.
COLUMN	??? Se copian las celdas con el símbolo ???.
CV_FOLDS	Los signos de interrogación (???) se tratan como una clase separada.
DELAY	??? Se copian las celdas con el símbolo ???.
FREQ	Los signos de interrogación (???) se tratan como una clase separada. En una columna numérica, el símbolo ??? se cuenta como un cero.
EXTRACT	Los signos de interrogación (???) en la predicate_col se tratan como ceros.
LAG	??? Se copian las celdas con el símbolo ???.
MERGE	??? Se copian las celdas con el símbolo ???.
SORT	Todas las celdas con el símbolo ??? se ordenan al final.
SAMPLE_RANDOM	??? Las celdas con el símbolo ??? pueden muestrearse.

Función de macro	Cómo se utilizan las celdas con el símbolo ???.
SELECT	??? Se copian las celdas con el símbolo ???.
SUBSAMPLE	??? Las celdas con el símbolo ??? pueden muestrearse.

### Funciones de macro que no pueden procesar celdas con el símbolo ???

Función de macro	Argumentos que no pueden contener el símbolo ???
GRID	col1, col2
HISTOGRAM	data, bin_col
INTEGRAL	data, multiplier
ISERROR	data

---

## Capítulo 4. IBM Referencia de macros

Esta sección proporciona información de referencia para las funciones de macro de hoja de cálculo disponibles en la suite de IBM Marketing Platform.

Las macros disponibles se listan en las siguientes páginas, en orden alfabético. Cada macro aparece con su sintaxis, parámetros posibles y algunos ejemplos.

**Importante:** No utilice nombres función nombres o palabras clave del Lenguaje de macros IBM para cabeceras de columna en tablas de usuario en IBM Campaign, ya sea que la correlación sea con una base de datos o con un archivo sin formato. Estas palabras reservadas pueden provocar errores si se utiliza en cabeceras de columna en tablas correlacionadas.

---

### ABS

#### Sintaxis

ABS(data)

#### Parámetros

data

Los valores numéricos para los que calcular el valor absoluto. Este parámetro puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a cualquiera de estos tipos. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para su producto.IBM

ABS calcula el valor absoluto de los números en el rango de datos especificado. El valor absoluto de un número es su valor sin el signo (es decir, los números positivos se mantienen como tales, los números negativos se devuelven como positivos). ABS devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene el valor absoluto de los números en la columna de entrada correspondiente.

#### Ejemplos

TEMP = ABS(-3) o TEMP = ABS(3)
Crea una columna denominada TEMP que contiene el valor 3.
TEMP = ABS(V1)
Crea una columna denominada TEMP, donde cada valor es el valor absoluto del contenido de la columna V1.
TEMP = ABS(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los valores absolutos del contenido de la columna V1, los valores de la columna VX son los valores absolutos del contenido de la columna V2, y los valores de la columna VY son los valores absolutos del contenido de la columnaV3.

TEMP = ABS(V1[10:20])

Crea una columna denominada TEMP, donde las primeras 11 celdas contienen los valores absolutos de los valores en las filas 10-20 de la columna V1. Las otras celdas de TEMP están vacías.

TEMP = ABS(V1[1:5]:V2)

Crea dos nuevas columnas denominadas TEMP y VX, con valores en las filas 1-5 de cada una (las otras celdas están vacías). Los valores en la columna TEMP son los valores absolutos de las filas correspondientes de la columna V1, y los valores en la columna VX son los valores absolutos de las filas correspondientes de la columna V2.

## Funciones relacionadas

Función	Descripción
SIGN	Calcula el signo (positivo o negativo) de los valores del rango de datos especificado

---

## ACOS

### Sintaxis

ACOS( data [, units\_keyword ])

### Parámetros

data

Los valores numéricos para calcular el valor de coseno de arco. Este parámetro puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a cualquiera de estos tipos. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

units\_keyword

Esta palabra clave opcional determina si los valores de entrada y los resultados se interpretan como grados o radianes. Seleccione una de las opciones siguientes:

radian - realiza los cálculos en radianes (valor predeterminado)

DEGREE - realiza los cálculos en grados

Si no se especifica este parámetro, el valor predeterminado es radianes. (Para convertir de radianes a grados, divida por PI y multiplíquelo por 180.)

Para obtener más detalles sobre la utilización de palabras clave en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

## Descripción

ACOS calcula el arco coseno de los valores en el rango de datos especificado. El arcocoseno es el ángulo cuyo coseno es el contenido de cada celda. ACOS devuelve una nueva columna para columna de entrada, cada una de las cuales contiene el arcocoseno de los números en la columna de entrada correspondiente.

Si se utiliza la palabra clave RADIAN, ACOS devuelve valores en el rango de 0 a  $\pi$ . Si se utiliza la palabra clave DEGREE, ACOS devuelve valores en el rango 0 - 180.

**Nota:** El contenido de cada celda de la columna especificada debe tener valores entre -1,0 y 1,0 inclusive. De lo contrario, devuelve una celda en blanco para cada entrada no válida.

## Ejemplos

TEMP = ACOS(0) o TEMP = ACOS(0, 0) o TEMP = ACOS(0, RADIAN)
Crea una columna denominada TEMP que contiene el valor 1.571 ( $\pi/2$ radianes).
TEMP = ACOS(0, 1) o TEMP = ACOS(0, DEGREE)
Crea una columna denominada TEMP que contiene el valor 90 (grados).
TEMP = ACOS(V1)
Crea una columna denominada TEMP, donde cada valor es el arcocoseno (expresado en radianes) del contenido de la columna V1.
TEMP = ACOS(V1:V3, 1)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los arcocosenos del contenido de la columna V1, los valores de la columna VX son los arcocosenos del contenido de la columna V2, y los valores de la columna VY son los arcocosenos del contenido de la columna V3. Todos los valores se expresan en grados.
TEMP = ACOS(V1[10:20])
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los arcocosenos de los valores en las filas 10-20 de la columna V1 (expresados en radianes). Las otras celdas de TEMP están vacías.
TEMP = ACOS(V1[1:5]:V2)
Crea dos columnas nuevas denominadas TEMP y VX, cada una con valores en las filas 1 a 5 (las otras celdas están vacías). Los valores en la columna TEMP son los arcocosenos de las filas correspondientes de la columna V1, y los valores en la columna VX son los arcocosenos de las filas correspondientes de la columna V2. Todos los valores se expresan en radianes.

## Funciones relacionadas

Función	Descripción
ACOT	Calcula la arcotangente del contenido del rango de datos especificado
ASIN	Calcula el arcocoseno del contenido del rango de datos especificado
ATAN	Calcula la arcotangente del contenido del rango de datos especificado
COS	Calcula el número coseno del contenido del rango de datos especificado

---

## ACOT

### Sintaxis

ACOT(data [, units\_keyword])

### Parámetros

data

Los valores numéricos para calcular el valor del arcocotangente. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

units\_keyword

Esta palabra clave opcional determina si los valores de entrada y los resultados se interpretan como grados o radianes. Seleccione una de las opciones siguientes:

RADIAN - Realiza los cálculos en radianes (valor predeterminado)

DEGREE - Realiza los cálculos en grados

Si no se especifica este parámetro, el valor predeterminado es radianes. (Para convertir de radianes a grados, divida por PI y multiplíquelo por 180.)

Para obtener más detalles sobre la utilización de palabras clave en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

### Descripción

ACOT devuelve el ángulo cuya cotangente es el contenido de cada celda. ACOT devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene el arcocotangente de los números en la columna de entrada correspondiente. Se utilizan número de coma flotante de 64-bit.

### Ejemplos

TEMP = ACOT(0.5) o TEMP = ACOT(0.5, 0) o TEMP = ACOT(0.5, RADIAN)
Crea una columna denominada TEMP que contiene el valor 2.157 (radianes).
TEMP = ACOT(1, 1) o TEMP = ACOT(1, DEGREE)
Crea una columna denominada TEMP que contiene el valor 0.022 (1/45) grados.
TEMP = ACOT(0)
Crea una columna denominada TEMP que contiene el valor MAX32_Float expresado en radianes.



TEMP = ACOT(V1)
Crea una nueva columna denominada TEMP, donde cada valor es el arcocotangente (expresado en radianes) del contenido de la columna V1.
TEMP = ACOT(V1:V3, 1)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los arcocotangentes del contenido de la columna V1, los valores de la columna VX son los arcocotangentes del contenido de la columna V2, y los valores de la columna VY son los arcocotangentes del contenido de la columna V3. Todos los valores se expresan en grados.
TEMP = ACOT(V1[10:20])
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los valores en las filas 10-20 de la columna V1 (expresados en radianes). Las otras celdas de TEMP están vacías.
TEMP = ACOT(V1[1:5]:V2)
Crea dos nuevas columnas denominadas TEMP y VX, con valores en las filas 1-5 (las otras celdas están vacías). Los valores en la columna TEMP son los arcocotangentes de las filas correspondientes de la columna V1, y los valores en la columna VX son los arcocotangentes de las filas correspondientes de la columna V2. Todos los valores se expresan en radianes.

## Funciones relacionadas

Función	Descripción
ACOS	Calcula el arcocoseno del contenido en el rango de datos especificado
ASIN	Calcula el arcoseno del contenido del rango de datos especificado
ATAN	Calcula la arcotangente del contenido del rango de datos especificado
COT	Calcula la cotangente del contenido del rango de datos especificado

---

## ADD\_MONTHS

### Sintaxis

ADD\_MONTHS(months, date\_string [, input\_format])

### Parámetros

months

Un entero que representa un número de meses a sumar a la date\_string.

date\_string

La fecha se devuelve en el formato predeterminado (DDMMMYYYY, o el formato especificado por el opcional input\_format argumento.

input\_format

El formato que se utilizará para la fecha calculada. Para obtener una lista de los formatos de fecha soportados, consulte la función DATE\_FORMAT. Tenga en cuenta que el formato de entrada determina tanto el formato de la serie de entrada y también el formato de la salida de la serie.

## Descripción

ADD\_MONTHS devuelve una fecha después de añadir el número de meses especificado en el date\_string especificado. La fecha se devuelve en el formato predeterminado (DDMMYYYY) o el formato especificado por el opcional input\_format argumento. Si desea un formato diferente de salida de datos, utilice DATE\_FORMAT.

Si aumentar el mes por el número de meses especificado genera una fecha válida, el resultado se calcula para que sea el último día del mes, como se muestra en el ejemplo siguiente semana. Cuando sea necesario, años bisiestos se tienen en cuenta. Por ejemplo, sumar un mes a 31-Ene-2012 dará como resultado en 29-Feb-2012.

## Ejemplos

ADD\_MONTHS(12, '25.06.2011') añade un año (12 meses) a la fecha especificada y devuelve la fecha 25.06.2012.

ADD\_MONTHS(3, '2011-06-25', DT\_DELIM\_Y\_M\_D) añade tres meses a la fecha especificada y devuelve la fecha 2011-09-25.

ADD\_MONTHS(1, '28.02.2011') devuelve la fecha 28.03.2011.

ADD\_MONTHS(1, '31.03.2012') devuelve la fecha 30.04.2012.

## Funciones relacionadas

Función	Descripción
DATE	Convierte una cadena de fecha en una fecha juliana.
DATE_FORMAT	Transforma una fecha de input_format en formato_salida.

---

## ALIGN



### Sintaxis

ALIGN(ref\_series, series, range)

### Parámetros

ref\_series

series

range

## Descripción

Alinea un rango con posición identificada por su serie a la serie de referencia en cada fila en <range> tiene un número correspondiente, como se identifica en <series>. La fila es alineada la fila de desplazamiento correcta uniendo <series> para el desplazamiento en <reference\_series>. Si <series> no contiene el número en <reference\_series>, se rellena con 0. Si <reference\_series> no contiene el número en <series>, la fila se ignora. Si más de 1 fila se alinea con la serie de referencia, la primera vez que se utiliza. Este se utiliza para alinear las fechas, tal como está expresado por números, asociados con serie de tiempo a partir de distintas fechas. De este modo, podemos generar un bloque donde cada fila son datos desde el mismo día.

---

## AND

### Sintaxis

```
data1 AND data2 data1 && data2
```

### Parámetros

data1

Los enteros no negativos para AND a nivel de bit con los valores en data2. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

data2

El número(s) de lógica AND con los valores de datos1. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en data2 debe ser igual al número de columnas en data1, a menos que data2 sea una constante. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

AND calcula el AND lógico entre dos rangos de datos especificados. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene el AND-ed lógico en data1 para la correspondiente columna data2 (es decir, la primera columna de data1 es AND-ed lógico con la primera columna de data, la segunda con la segunda, etc).

Si data2 es una constante, cada valor en data1 es AND-ed lógico por ese valor. Si data2 contiene una o más columnas, los cálculos se realizan fila por fila entre una columna de data1 y una columna de data2. La primera fila de data1 es AND-ed lógico al valor de la primera fila de data2, la segunda con la segunda, etc. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** El operador AND puede ser abreviado con un signo( &&). Utilice el doble signo para separar los dos argumentos (por ejemplo, para especificar V1 AND 3, simplemente escriba V1&&3).

Este macro está disponible en IBM Interact.

## Ejemplos

<code>TEMP = 1 AND 8</code> o <code>TEMP = 1 &amp;&amp; 8</code>
Crea una nueva columna denominada TEMP que contiene el valor uno (cualquier número distinto de cero es tratado como un uno).
<code>TEMP = V1 &amp;&amp; 1</code>
Crea una nueva columna denominada TEMP con el valor uno para cada valor de la columna V1.
<code>TEMP = V1 &amp;&amp; V1</code>
Crea una nueva columna denominada TEMP con el valor uno para cada valor distinto de cero en la columna V1 y el valor cero para cada cero en la columna V1.
<code>TEMP = V1 &amp;&amp; V2</code>
Crea una nueva columna denominada TEMP, donde cada valor es el valor AND-ed lógico de la fila en la columna V1 con la fila correspondiente de la columna V2.
<code>TEMP = V1:V3 &amp;&amp; V4:V6</code>
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores AND-ed lógico de la columna V1 con los correspondientes valores en la fila de la columna V4. La columna VX contiene los valores AND-ed lógico de las columnas V2 y V5. La columna VY contiene los valores AND-ed lógico de las columnas V3 y V6.
<code>TEMP = V1[10:20] &amp;&amp; V2</code> o <code>TEMP = V1[10:20] &amp;&amp; V2[1:11]</code>
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los valores AND-ed lógico resultantes de los valores en las filas 10-20 de la columna V1 por los valores en las filas 1-11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
NOT	Calcula el número NOT lógico del contenido del rango de datos rango de datos especificado
OR	Calcula el OR lógico entre dos rangos de datos especificados

---

## ASIN

### Sintaxis

`ASIN(data [, units_keyword])`

### Parámetros

data

Los valores numéricos para calcular el valor de seno de arco. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a

uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

units\_keyword

Esta palabra clave opcional determina si los valores de entrada y los resultados se interpretan como grados o radianes. Seleccione una de las opciones siguientes:

RADIAN - Realiza los cálculos en radianes (valor predeterminado)

DEGREE - Realiza los cálculos en grados

Si no se especifica este parámetro, el valor predeterminado es radianes. (Para convertir de radianes a grados, divida por PI y multiplíquelo por 180.)

Para obtener más detalles sobre la utilización de palabras clave en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

## Descripción

ASIN calcula el arcoseno de los valores en el rango de datos especificado. El arcoseno es el ángulo cuyo seno es el contenido de cada celda. ASIN devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene el arcoseno de los números en la columna de entrada correspondiente.

Si se utiliza la palabra clave RADIAN, ASIN devuelve valores en el rango  $-\pi/2$  a  $\pi/2$ . Si se utiliza la palabra clave DEGREE , ASIN devuelve valores en el rango -90 a 90.

**Nota:** El contenido de cada celda de la columna especificada debe tener valores entre -1,0 y 1,0 inclusive. De lo contrario, devuelve ??? para cada entrada no válida.

## Ejemplos

TEMP = ASIN(0.5) o TEMP = ASIN(0.5, 0) o TEMP = ASIN(0.5, RADIAN)
Crea una nueva columna denominada TEMP que contiene el valor 0.524 ( $\pi/6$ radianes).
TEMP = ASIN(0.5, 1) o TEMP = ASIN(0.5, DEGREE)
Crea una nueva columna denominada TEMP que contiene el valor 30 (grados).
TEMP = ASIN(V1)
Crea una nueva columna denominada TEMP, donde cada valore es el arcoseno (expresado en radianes) del contenido de la columna V1.
TEMP = ASIN(V1:V3, 1)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los arcosenos del contenido de la columna V1, los valores en la columna VX son los arcosenos del contenido de la columna V2, y los valores en la columna VY son los arcosenos del contenido de la columna V3. Todos los valores se expresan en grados.

<p>TEMP = ASIN(V1[10:20])</p> <p>Crea una nueva columna denominada TEMP, donde las 11 primeras celdas contienen los arcosenos de los valores en las filas 10-20 de la columna V1 (expresados en radianes). Las otras celdas de TEMP están vacías.</p>
<p>TEMP = ASIN(V1[1:5]:V2)</p> <p>Crea dos nuevas columnas denominadas TEMP y VX, con valores en las filas 1-5 (las otras celdas están vacías). Los valores en la columna TEMP son los arcosenos de las filas correspondientes de la columna V1, y los valores en la columna VX son los arcosenos de las filas correspondientes en la columna V2. Todos los valores se expresan en radianes.</p>

## Funciones relacionadas

Función	Descripción
ACOS	Calcula el arcoseno del contenido en el rango de datos especificado
ACOT	Calcula la arcotangente del contenido del rango de datos especificado
ATAN	Calcula la arcotangente del contenido del rango de datos especificado
SIN	Calcula el seno del contenido del rango de datos especificado

---

## ATAN

### Sintaxis

ATAN(data [, units\_keyword])

### Parámetros

data

Los valores numéricos para calcular el valor de la arcotangente. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

units\_keyword

Esta palabra clave opcional determina si los valores de entrada y los resultados se interpretan como grados o radianes. Seleccione una de las opciones siguientes:

RADIAN - Realiza los cálculos en radianes (valor predeterminado)

DEGREE - Realiza los cálculos en grados

Si no se especifica este parámetro, el valor predeterminado es radianes. (Para convertir de radianes a grados, divida por PI y multiplíquelo por 180.)

Para obtener más detalles sobre la utilización de las palabras claves en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte “Especificaciones de formato” en la página 27.

## Descripción

ATAN calcula la arcotangente de los valores en el rango de datos especificado. El arcotangente es el ángulo cuya tangente es el contenido de cada celda. ATAN devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene la arcotangente de los números en la columna de entrada correspondiente.

Si se utiliza la palabra clave RADIAN , ATAN devuelve valores en el rango  $-\pi/2$  a  $\pi/2$ . Si se utiliza la palabra clave DEGREE , ATAN devuelve valores en el rango -90 a 90.

## Ejemplos

TEMP = ATAN(1) o TEMP = ATAN(1, 0) o TEMP = ATAN(1, RADIAN)
Crea una nueva columna denominada TEMP que contiene el valor 0.785 ( $\pi/4$ radianes).
TEMP = ATAN(1, 1) o TEMP = ATAN(1, DEGREE)
Crea una nueva columna denominada TEMP que contiene el valor 45 (grados).
TEMP = ATAN(V1)
Crea una nueva columna denominada TEMP, donde cada valor es arcotangente (expresado en radianes) del contenido de la columna V1.
TEMP = ATAN(V1:V3, 1)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los arcotangentes del contenido de la columna V1, los valores en la columna VX son los arcotangentes del contenido de la columna V2, y los valores en la columna VY son los arcotangentes del contenido de la columna V3. Todos los valores se expresan en grados.
TEMP = ATAN(V1[10:20])
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los arcotangentes de los valores en las filas 10-20 de la columnaV1 (expresados en radianes). Las otras celdas de TEMP están vacías.
TEMP = ATAN(V1[1:5]:V2)
Crea dos nuevas columnas denominadas TEMP y VX, con los valores en la filas 1-5 (las otras celdas están vacías). Los valores en la columna TEMP son los arcotangentes de las filas correspondientes de la columna V1, y los valores en la columna VX son los arcotangentes de las filas correspondientes de la columna V2. Todos los valores se expresan en radianes.

## Funciones relacionadas

Función	Descripción
ACOS	Calcula el número NOT lógico del contenido del rango de datos especificado arcocoseno del contenido del rango de datos especificado
ASIN	Calcula el número NOT lógico del contenido del rango de datos especificado arcoseno del contenido del rango de datos especificado

Función	Descripción
ATAN	Calcula la arcotangente del contenido del rango de datos especificado
TAN	Calcula la tangente del contenido del rango de datos especificado

---

## AVG

### Sintaxis

AVG(data [, keyword])

### Parámetros

data

Los valores numéricos de los que calcular la media aritmética. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las opciones siguientes:

ALL - realiza el cálculo para todas la celdas en data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de data

Para obtener más detalles sobre la utilización de palabras clave en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabrasclave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

**Nota:** Varias funciones de macro toman los parámetros de la palabra clave {ALL | COL | ROW}. Estas palabras clave no se aplican en **IBM Campaign** porque los datos de entrada son siempre una columna o un campo único. La macro siempre se comportará como si se especificara la palabra clave COL. Por lo tanto, no necesita especificar estas palabras claves cuando utilice **IBM Campaign** .

### Descripción

AVG calcula el promedio aritmético o la media de las celdas en un rango de datos especificado. La media aritmética se calcula sumando el contenido de todas las celdas dividido por el número de celdas. El número de columnas devueltas por AVG depende de keyword.

- Si keyword es ALL, AVG devuelve una nueva columna que contiene un valor único (el promedio de todas las celdas en data).



- Si keyword es COL, AVG devuelve una nueva columna para cada columna de entrada. Cada columna nueva contiene un valor (el promedio de todas las celdas de la correspondiente columna de entrada).
- Si keyword es ROW, AVG devuelve una nueva columna que contiene el promedio de todas las filas de data.

**Nota:** Las celdas en blanco se pasan por alto en el cálculo.

**Nota:** AVG es lo mismo que la función de macro MEAN.

Esta macro está disponible en IBM Interact.

## Ejemplos

TEMP = AVG(MERGE(3, 4, 5)) o TEMP = AVG(MERGE(3, 4, 5), ALL)
Crea una nueva columna denominada TEMP que contiene el valor 4.
TEMP = AVG(MERGE(-10, 3, 10))Crea una nueva columna denominada TEMP que contiene el valor 1.
TEMP = AVG(V1) Crea una nueva columna denominada TEMP que contiene un valor único ue el promedio aritmético del contenido de la columna V1.
TEMP = AVG(V1:V3) Crea una nueva columna denominada TEMP que contiene un único valor que es el promedio aritmético del contenido en las columnas V1, V2, y V3.
TEMP = AVG(V1[10:20]) Crea una nueva columna denominada TEMP que contiene un valor único que es el promedio aritmético de de las celdas en las filas 10-20 de la columna V1.
TEMP = AVG(V1[1:5]:V4) Crea una nueva columna denominada TEMP que contiene un valor único que es el promedio aritmético de las celdas en las filas 1-5 de las columnas V1 a V4.
TEMP = AVG(V1:V3, COL) Crea tres nuevas columnas denominadas TEMP, VX, y VY. El valor único en la columna TEMP es el promedio aritmético del contenido de la columna V1, el valor único en la columna VX es el promedio aritmético del contenido de la columna V2, y el valor único en la columna VY es el promedio aritmético del contenido de la columna V3.
TEMP = AVG(MERGE(1,4), COL) Crea dos nuevas columnas denominadas TEMP y VX. TEMP contiene un valor único de uno; VX contiene un valor único de cuatro.
TEMP = AVG(V1[1:5]:V3, COL) Crea tres nuevas columnas denominadas TEMP, VX, y VY, cada una contiene un valor único.El valor en la columna TEMP es el promedio aritmético de las celdas en la filas 1-5 de la columna V1, el valor en la columna VX es el promedio aritmético de las celdas en las filas 1-5 de la columna V2, y el valor en la columna VY es el promedio aritmético de las celdas en las filas 1-5 de la columna V3.
TEMP = AVG(V1, ROW) Crea una nueva columna denominada TEMP, que contiene los mismos valores que la columna V1 (el promedio aritmético de cada número es el mismo número).
TEMP = AVG(V1:V3, ROW) Crea una nueva columna denominada TEMP donde cada entrada es el promedio aritmético de la fila correspondiente en las columnas V1, V2, y V3.
TEMP = AVG(V1[1:5]:V3, ROW) Crea una nueva columna denominada TEMP, donde las celdas en las filas 1-5 contienen el promedio aritmético de las filas correspondientes en las columnas V1 a V3. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
AVG_DEV	Calcula la desviación promedio de un rango de celdas
SUM o TOTAL	Calcula la suma de un rango de celdas

---

## AVG\_DEV



### Sintaxis

AVG\_DEV(data [, keyword])

### Parámetros

data

Los valores para calcular la desviación promedio. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las opciones siguientes:

ALL - Realiza el cálculo en todas las celdas de data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de data

Para obtener más detalles sobre la utilización de las palabras claves en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

### Descripción

AVG\_DEV calcula la desviación promedio de los valores en el rango de datos especificados desde el promedio. El promedio aritmético es calculado sumando el contenido de todas las celdas y dividiendo entre el número de celdas. La ecuación completa para AVG\_DEV es la siguiente:

$$\sum_i^n |x_i - \text{mean}|$$

El número de columnas devueltas por AVG depende de keyword.

- Si keyword es ALL, AVG devuelve una nueva columna que contiene un valor único (la desviación promedio de todas las celdas en data).
- Si keyword es COL, AVG devuelve una nueva columna para cada columna de entrada. Cada nueva columna contiene un valor (la desviación promedio de todas las celdas en la columna de entrada correspondiente).
- Si keyword es ROW, AVG devuelve una nueva columna que contiene la desviación promedio a través de cada fila de data.

**Nota:** Las celdas en blanco se ignoran en el promedio.

## Ejemplos

TEMP = AVG_DEV(MERGE(3, 4, 5)) o TEMP = AVG_DEV(MERGE(3, 4, 5), ALL)
Crea una nueva columna denominada TEMP que contiene el valor 0.67.
TEMP = AVG_DEV(COLUMN(-4, 0))
Crea una nueva columna denominada TEMP que contiene el valor 2.
TEMP = AVG_DEV(V1)
Crea una nueva columna denominada TEMP que contiene un valor único que es la desviación promedio del contenido de la columna V1.
TEMP = AVG_DEV(V1:V3)
Crea una nueva columna denominada TEMP que contiene un valor único que es la desviación promedio del contenido de las columnas V1, V2, y V3.
TEMP = AVG_DEV(V1[10: 20])
Crea una nueva columna denominada TEMP que contiene un valor único que es la desviación promedio de las celdas en las filas 10-20 de la columna V1.
TEMP = AVG_DEV(V1[1:5]:V4)
Crea una nueva columna denominada TEMP que contiene un valor único que es la desviación promedio de las celdas en las filas 1-5 de las columnas V1 a V4.
TEMP = AVG_DEV(V1:V3, COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY. El valor único en la columna TEMP es la desviación promedio del contenido de la columna V1, el valor único en la columna VX es la desviación promedio del contenido de la columna V2, y el valor único en la columna VY es la desviación promedio del contenido de la columna V3.
TEMP = AVG_DEV(MERGE(1,4), COL)
Crea dos nuevas columnas denominadas TEMP y VX, ambas contienen el valor único 0.

<p>TEMP = AVG_DEV(V1[1:5]:V3, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene un valor único. El valor en la columna TEMP es la desviación promedio de las celdas en las filas 1-5 de la columna V1, el valor en la columna VX es la desviación promedio de las celdas en las filas 1-5 de la columna V2, y el valor en la columna VY es la desviación promedio de las celdas en las filas 1-5 de la columna V3.</p>
<p>TEMP = AVG_DEV(V1, ROW)</p> <p>Crea una nueva columna denominada TEMP, que contiene un cero por cada valor de la columna V1 (la desviación promedio para cualquier número es cero).</p>
<p>TEMP = AVG_DEV(V1:V3, ROW)</p> <p>Crea una nueva columna denominada TEMP donde cada entrada es la desviación promedio de la fila correspondiente en las columnas V1, V2, y V3.</p>
<p>TEMP = AVG_DEV(V1[1:5]:V3, ROW)</p> <p>Crea una nueva columna denominada TEMP, donde las celdas en las filas 1-5 contienen la desviación promedio de la fila correspondiente a través de las columnas V1 a V3. Las otras celdas de TEMP están vacías.</p>

## Funciones relacionadas

Función	Descripción
AVG o MEAN	Calcula la media aritmética o el promedio de un rango de celdas
SUM o TOTAL	Calcula la suma de un rango de celdas

---

## BETWEEN

### Sintaxis

value1 BETWEEN value2 AND value3

### Parámetros

Equivalent to value1 >= value2 AND < value3

### Descripción

BETWEEN es una variante especial del predicado de comparación. Los detalles de este predicado son importantes y el orden de los operandos tiene algunas implicancias inesperadas. Vea la sección de ejemplos.

**Nota:** FROM y FOR utilizan sintaxis idéntica.

### Ejemplos

<pre>SELECT * FROM movie_titles WHERE our_cost BETWEEN 11.00 and 27.50 ;</pre> <p>devuelve una lista de las películas que se pueden comprar por no menos de \$11,00, pero su coste es de no mas de 27,50.</p>
---

10 BETWEEN 5 AND 15 Is true, but: 10 BETWEEN 15 AND 5 Is false:

la manera equivalente de expresar BETWEEN (utilizar AND) tiene un orden específico que no es relevante cuando se utilizan literales, pero es bastante relevante cuando proporciona value2 y value3 utilizando variables host, parámetros o incluso subconsultas.

---

## BIT\_AND

### Sintaxis

```
data1 BIT_AND data2 data1 & data2
```

### Parámetros

data1

Los enteros no negativos para AND a nivel de bit con los valores en data2. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

data2

Los enteros no negativos para AND a nivel de bit con los valores en data1. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en data2 debe ser igual al número de columnas en data1, a menos que data2 sea una constante. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

BIT\_AND realiza el AND a nivel de bit entre dos rangos de datos especificados. Devuelve una nueva columna para cada columna de entrada, cada una contiene la columna correspondiente en data1 AND-ed a nivel de bit a la columna correspondiente de data2 (es decir, la primera columna de data1 es AND-ed a nivel de bit para la primera columna de data, la segunda columna con la segunda columna, etc.).

Si data2 es una constante, cada valor en data1 es AND-ed a nivel de bit para ese valor. Si data2 contiene una o más columnas, los cálculos se realizan fila por fila entre una columna de data2 y una columna de data1. La primera fila de data1 es AND-ed a nivel de bit para el valor de la primera fila de data2, la segunda fila con la segunda fila, etc. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** La precisión de esta función de macro se limita a valores enteros menores que  $2^{24}$  . No se permiten valores negativos.

**Nota:** El operador BIT\_AND puede ser abreviado con un signo (&). Utilice el signo para separar los dos argumentos (por ejemplo, para especificar: se the ampersand to separate the two arguments (for example, to specify BIT\_AND(V1, 3), simplemente escriba V1&3.

## Ejemplos

<code>TEMP = 3 BIT_AND 7</code> o <code>TEMP = 3 &amp; 7</code>
Crea una nueva columna denominada TEMP que contiene el valor tres (AND a nivel de bit de 011 y 111 equivale a 011).
<code>TEMP = V1 &amp; 8</code>
Crea una nueva columna denominada TEMP, donde cada valor es el AND-ed a nivel de bit del contenido de la columna V1 con el valor binario 1000.
<code>TEMP = V1 &amp; V1</code>
Crea una nueva columna denominada TEMP que contiene los mismos contenidos de la columna V1 (cada valor AND-ed con sí mismo devuelve su mismo valor).
<code>TEMP = V1 &amp; V2</code>
Crea una nueva columna denominada TEMP, donde cada valor es el valor AND-ed a nivel de bit de la fila de la columna V1 con el correspondiente valor de fila de la columna V2.
<code>TEMP = V1:V3 &amp; V4:V6</code>
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores en V1 bitwise AND-ed with the corresponding row values of column V4. La columna VX contiene los valores AND-ed a nivel de bit para las columnas V2 y V5. La columna VY contiene los valor AND-ed a nivel de bit para las columnas V3 y V6.
<code>TEMP = V1[10:20] &amp; V2</code> o <code>TEMP = V1[10:20] &amp; V2[1:11]</code>
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen el resultado AND-ed a nivel de bit para los valores en la filas 10-20 de la columna V1 por los valores en las filas 1-11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
<code>BIT_NOT</code>	Calcula NOT a nivel de bit del contenido del rango de datos especificado
<code>BIT_OR</code>	Calcula OR a nivel de bit entre dos rangos de datos especificados
<code>BIT_XOR</code> o <code>XOR</code>	Calcula XOR a nivel de bit entre dos rangos de datos especificados

---

## BIT\_NOT

### Sintaxis

`BIT_NOT data ~ data`

### Parámetros

`data`

Los enteros no negativos para NOT a nivel de bit. Este puede ser un valor constante, una columna. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

BIT\_NOT calcula el NOT a nivel de bit de los valores en el rango de datos especificado. Devuelve una nueva columna para cada columna de entrada, cada una contiene el NOT a nivel de bit de los valores en las columnas correspondientes de data.

**Nota:** La precisión de esta función de macro se limita a valores enteros menores que  $2^{24}$ . No se permiten valores negativos.

**Nota:** Utilización de una columna que contiene el mismo número  $x$  en cada fila como data es lo mismo que utilizar la constant  $x$  como data.

**Nota:** El operador BIT\_NOT puede abreviarse con un tilde (~). Utilice el tilde antes del valor de datos (por ejemplo, para especificar BIT\_NOT(V1), puede simplemente escribir ~V1.

## Ejemplos

TEMP = BIT\_NOT 3 o TEMP = ~3

Crea una nueva columna denominada TEMP que contiene el valor de cuatro (NOT a nivel de bit de 011 igual 100).

TEMP = ~V1

Crea una nueva columna denominada TEMP, donde cada valor es el NOT a nivel de bit del contenido de la columna V1.

TEMP = ~V1:V3

Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los NOT a nivel de bit del contenido de la columna V1, los valores de la columna VX son los NOT a nivel de bit del contenido de la columna V2, y los valores de la columna VY son los NOT a nivel de bit del contenido de la columna V3.

TEMP = ~V1[100:200]

Crea una nueva columna denominada TEMP, donde las primeras 101 celdas contienen el NOT a nivel de bit de los valores en las filas 1-50 de la columna V1.

## Funciones relacionadas

Función	Descripción
BIT_AND	Calcula AND a nivel de bit entre dos rangos de datos especificados
BIT_OR	Calcula OR a nivel de bit entre dos rangos de datos especificados
BIT_XOR o XOR	Calcula XOR a nivel de bit entre dos rangos de datos especificados

---

## BIT\_OR

### Sintaxis

data1 BIT\_OR data2 data1 OR data2 data1 | data2

## Parámetros

data1

Los enteros no negativos de bit OR con los valores de data2. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

data2

El entero no negativo (s) de bit OR con los valores de data1. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en data2 debe ser igual al número de columnas en data1, a menos que data2 sea una constante. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

BIT\_OR realiza a nivel de bit OR entre los dos rangos de datos especificados. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene la correspondiente columna en data1 a nivel de bit OR-ed a la correspondiente columna de data2 (es decir, la primera columna de data1 es a nivel de bit OR-ed a la primera columna de data, la segunda columna con la segunda columna, etc).

Si data2 es una constante, cada valor en data1 es a nivel de bit OR-ed por ese valor. Si data2 contiene una o más columnas, los cálculos se realizan en una base fila por fila entre una columna de data1 y una columna de data2. La primera fila a nivel de bit OR-ed es data1 al primer valor de fila de data2, la segunda fila con la segunda fila, etc. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** La precisión de esta función de macro se limita a los valores enteros menores que  $2^{24}$  . No se permiten valores negativos.

**Nota:** El operador BIT\_OR puede ser abreviado con una barra vertical (|). Utilice la barra vertical para separar las dos columnas (por ejemplo, para especificar BIT\_OR(V1, 3), puede simplemente el tipo V1|3. También puede utilizar OR.

## Ejemplos

```
TEMP = 3 BIT_OR 7 o TEMP = 3 OR 7 o TEMP = 3 | 7
```

Crea una nueva columna de nombre TEMP que contiene el valor de siete (a nivel de bit OR de 011 y 111 igual a 111).

```
TEMP = V1 | 8
```

Crea una nueva columna de nombre TEMP, donde cada valor es el contenido de la columna V1 a nivel de bit OR-ed con el valor binario 1000.

```
TEMP = V1 | V1
```

Crea una nueva columna de nombre TEMP que contiene el mismo contenido que la columna V1 (cada valor OR-ed en sí mismo produce el propio).



TEMP = V1   V2
Crea una nueva columna de nombre TEMP, donde cada valor es el valor de fila de la columna V1 a nivel de bit OR-ed con el correspondiente valor de fila de la columna V2.
TEMP = V1:V3   V4:V6
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los OR lógicos entre los valores en V1 y los correspondientes valores de fila de la columna V4. La columna VX contiene los OR lógicos entre los valores de las columnas V2 y V5. La columna VY contiene los OR lógicos entre los valores de las columnas V3 y V6.
TEMP = V1[10:20]   V2 o TEMP = V1[10:20]   V2[1:11]
Crea una nueva columna de nombre TEMP, donde las primeras 11 celdas contienen el resultado a nivel de bit OR-ed de los valores en las filas 10-20 de la columna V1 por los valores en las filas 1-11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
BIT_AND	Calcula AND a nivel de bit entre dos rangos de datos especificados
BIT_NOT	Calcula NOT a nivel de bit del contenido del rango de datos especificado
BIT_XOR o XOR	Calcula XOR a nivel de bit entre dos rangos de datos especificados

---

## BIT\_XOR

### Sintaxis

data1 BIT\_XOR data2

### Parámetros

data1

Los enteros no negativos para XOR a nivel de bit con los valores en data2. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para su producto de IBM .

data2

Los enteros no negativos para XOR a nivel de bit con los valores en data1. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en data2 debe ser igual al número de columnas en data1, a menos que data2 sea una constante. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para su producto de IBM .

## Descripción

BIT\_XOR efectúa una operación XOR a nivel de bit entre los dos rangos de datos especificados. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene la columna correspondiente en data1 a nivel de bit XOR-ed a la columna correspondiente de data2 (es decir, la primera columna de data1 es a nivel de bit XOR-ed a la primera columna de data, la segunda columna con la segunda columna, etc).

Si data2 es una constante, cada valor en data1 está unido por un XOR a nivel de bit por ese valor. Si data2 contiene una o más columnas, los cálculos se realizan fila por fila entre una columna de data1 y una columna de data2. La primera fila de data1 está unida por un XOR a nivel de bit al primer valor de fila de data2, la segunda fila, a la segunda fila y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** La precisión de esta función de macro se limita a los valores enteros menores que  $2^{24}$ . No se permiten valores negativos.

## Ejemplos

TEMP = 3 BIT_XOR 7
Crea una nueva columna denominada TEMP que contiene el valor cuatro (XOR a nivel de bit de 011 y 111 es igual a 100).
TEMP = V1 BIT_XOR 8
Crea una nueva columna de nombre TEMP, donde cada valor es el contenido de la columna V1 unido por un XOR a nivel de bit con el valor binario 1000.
TEMP = V1 BIT_XOR V1
Crea una nueva columna denominada TEMP que contiene todos los ceros (cada valor unido por un XOR a sí mismo produce cero).
TEMP = V1 BIT_XOR V2
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 unido por un XOR a nivel de bit al valor de la fila correspondiente de la columna V2.
TEMP = V1:V3 BIT_XOR V4:V6
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores en V1 unidos por un XOR a nivel de bit a los valores de la fila correspondiente de la columna V4. La columna VX contiene los valores unidos por un XOR a nivel de bit de las columnas V2 y V5. La columna VY contiene los valores unidos por un XOR a nivel de bit de las columnas V3 y V6.
TEMP = V1[10:20] BIT_XOR V2 o TEMP = V1[10:20] BIT_XOR V2[1:11]
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen el resultado unido por el XOR a nivel de bit de los valores en las filas 10-20 de la columna V1 por los valores en las filas 1-11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
BIT_AND	Calcula AND a nivel de bit entre dos rangos de datos especificados

Función	Descripción
BIT_NOT	Calcula NOT a nivel de bit del contenido del rango de datos especificado
BIT_OR	Calcula OR a nivel de bit entre dos rangos de datos especificados

---

## BUFFER



### Sintaxis

BUFFER(data)

### Parámetros

data

Los valores para copiar como constantes. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

BUFFER hace una copia dinámica de los valores de rango de datos especificado. Estos valores copiados cambian si cambian los valores correspondientes en la entrada de la(s) columnas(s). BUFFER devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene una copia de los valores de la correspondiente columna de entrada.

**Nota:** Aplicación de la función de macro BUFFER puede resultar en importantes mejoras en el rendimiento ejecutando experimentos. Si la entrada o salida de las columnas para un experimento se basan en cálculos complejos (lentos) en la hoja de cálculo, añada la función de macro BUFFER para cada una de las columnas. Esto va a calcular los valores y almacenar los valores calculados. De lo contrario, cada vez que el experimento accede a la hoja de cálculo para patrones de datos, los valores deben ser recalculados. Si cambian algunos de los valores de entrada, los datos en BUFFER se actualizarán dinámicamente como otras funciones de macro en la hoja de cálculo.

**Nota:** Si no cambian nunca los valores de datos, en su lugar utilice la función de macro de CONSTANT . Esto crea una copia estática de los datos del rango.

**Nota:** Si una función de usuario se construye a partir de una definición de función utilizando la función de macro BUFFER, la parte de la definición de función encerrada en la función de macro BUFFER se considera una constante. No serán necesarias todas las variables de entrada para aplicar la función de usuario.

## Ejemplos

TEMP = BUFFER(4.3) Crea una nueva columna de nombre TEMP que contiene el valor 4.3.
TEMP = BUFFER(V1) Crea una nueva columna de nombre TEMP, donde cada valor es una copia del contenido de la columna V1.
TEMP = BUFFER(V1) Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son copias del contenido de la columna V1, los valores de la columna VX son copias del contenido de la columna V2, y los valores de la columna VY son copias del contenido de la columna V3.
TEMP = BUFFER(V1) Crea una nueva columna de nombre TEMP, donde las primeras 11 celdas contienen copias de los valores en las filas 10-20 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = BUFFER(V1) Crea dos nuevas columnas de nombre TEMP y VX, cada uno con valores en las filas 1-50 (las otras celdas están vacías). Los valores en la columna TEMP son copias de las filas en la columna V1, y los valores en la columna VX son copias de los valores en la columna V2.
TEMP = BUFFER(EXTRACT(!ISERROR(V1:V3), V1:V3)) Crea tres nuevas columnas denominadas TEMP, VX y VY. Las tres columnas contienen las filas válidas (es decir, filas que no contienen ninguna ??? celda) de columnas V1 - V3. La derivación de la función de macro EXTRACT dentro de CONSTANT impide que las columnas TEMP, VX y VY se vuelvan a calcular si las columnas V1 - V3 cambian, evitando así la intensa actividad de la función de macro EXTRACT.

## Funciones relacionadas

Función	Descripción
CONSTANT	Copia el rango de datos de entrada una vez (actualización no dinámica)

---

## CEILING

### Sintaxis

CEILING(data)

### Parámetros

data

Los valores numéricos para calcular el ceiling de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

CEILING calcula el ceiling de los valores en el rango de datos especificado. El ceiling de un número es el entero más pequeño *no* menor que el número. CEILING devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene el ceiling de números en la correspondiente columna de entrada.

## Ejemplos

TEMP = CEILING(4.3)
Crea una nueva columna denominada TEMP que contiene el valor 5.
TEMP = CEILING(2.9)
Crea una nueva columna denominada TEMP que contiene el valor -2.
TEMP = CEILING(V1)
Crea una nueva columna de nombre TEMP, donde cada valor es el ceiling del contenido de la columna V1.
TEMP = CEILING(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los ceilings del contenido de la columna V1, los valores de la columna VX son los ceilings del contenido de la columna V2, y los valores de la columna VY son los ceilings del contenido de la columna V3.
TEMP = CEILING(V1[10:20])
Crea una nueva columna de nombre TEMP, donde las primeras 11 celdas contienen los ceilings de los valores en las filas 10-20 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = CEILING(V1[50:99]:V2)
Crea dos nuevas columnas de nombre TEMP y VX, cada uno con valores en las filas 1-50 (las otras celdas están vacías). Los valores en la columna TEMP son los ceilings de las filas de la columna V1, y los valores en la columna VX son los ceilings de los valores en la columna V2.

## Funciones relacionadas

Función	Descripción
FLOOR o INT	Calcula el floor de cada valor del rango de datos especificado
FRACTION	Devuelve la parte fraccionaria de cada valor del rango de datos especificado
TRUNCATE	Devuelve la parte no fraccionaria de cada valor del rango de datos especificado

---

## COLUMNA

### Sintaxis

COLUMN(data [, data]...) or (data [, data]...)

### Parámetros

data

Un valor para utilizarlo en la creación de una columna. Esto puede ser un valor constante (numérico o de texto ASCII en comillas), una columna, un rango de celdas, o una expresión que se evalúa para cualquiera de los anteriores. Este parámetro puede repetirse varias veces, pero los parámetros deben tener la misma dimensionalidad (es decir, ancho de columna) como el primer parámetro. Todos los valores en todos los parámetros de data deben ser numéricos o texto ASCII (es decir, no puede mezclar numéricos y valores de texto). Si se proporcionan varios parámetros de data, todos ellos deben tener el mismo número de columnas. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM.

## Descripción

COLUMN concatena verticalmente sus entradas en columnas de una función de grupo. Devuelve el mismo número de nuevas columnas como en cada parámetro de entrada. Un número ilimitado de argumentos pueden ser proporcionados. Todos los argumentos deben ser numéricas o cadenas de texto ASCII (es decir, no puedo mezclar valores numéricos y de texto).

**Nota:** La función de macro COLUMN puede ser abreviado por el listado dedata argumentos separados por comas dentro de paréntesis (por ejemplo, (por ejemplo, TEMP = MEAN((1,2,3,4), ALL). Si no se utiliza dentro de otra función de macro, el par de paréntesis no es necesario (por ejemplo, V1=1,2,3 es equivalente a V1=COLUMN(1,2,3).

## Ejemplos

TEMP = COLUMN(3, 4, 5) o TEMP = (3,4,5) o TEMP = 3,4,5
Crea una nueva columna de nombre TEMP con las primeras tres celdas que contienen los valores 3, 4, y 5.
TEMP = COLUMN("one", "two", "three")
Crea una nueva columna de nombre TEMP con las primeras tres celdas que contienen los valores one, two, y three.
TEMP = AVG(V1), STDV(V1)
Crea una nueva columna de nombre TEMP con el promedio de la columna V1 en la primera celda, y la desviación estándar de la columna V1 en la segunda celda.
TEMP = V1:V2, V3:V4
Crea dos nuevas columnas de nombre TEMP y VX donde la columna TEMP contiene los valores de la columna V1 seguido de los valores de la columna V3. La columna VX contiene los valores de la columna V2 seguido de los valores de la columna V4.
TEMP = V1:V2, V3:V4
Crea dos nuevas columnas de nombre TEMP y VX donde la columna TEMP contiene los valores de las celdas 1-10 de la columna V1 seguido de todos los valores de la columna V3. La columna VX contiene los valores de las celdas 1-10 de la columna V2 seguido de todos los valores de la columna V4.
TEMP = V1:V2, V3:V4
Crea dos nuevas columnas de nombre TEMP y VX, cada uno de los cuales contiene un valor único. La columna TEMP contiene el promedio de las columnas V1 y V2. La columna VX contiene el promedio de las columnas V3 y V4.

## Funciones relacionadas

Función	Descripción
MERGE	Crea un grupo de datos concatenando horizontalmente los valores de entrada
TO	Generar operador de rango
TRANSPOSE	Traslada un rango de datos especificado

---

## CONSTANTE



### Sintaxis

CONSTANT(data)

### Parámetros

data

Los valores para copiar como constantes. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

CONSTANT hace una copia estática de los valores del rango de datos especificado. Estos valores copiados no *cambian* si cambian los valores correspondientes en las columnas de entrada. Los valores de datos se copian una vez en el momento en que se aplica la función de macro. CONSTANT devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene una copia estática de los valores en la correspondiente columna de entrada.

**Nota:** Aplicar la función de macro CONSTANT puede resultar en importantes mejoras en el rendimiento ejecutando experimentos. Si la entrada o salida de las columnas para un experimento se basan en cálculos complejos (lentos) en la hoja de cálculo, añada la función de macro CONSTANT para cada una de las columnas. Esto va a calcular los valores una vez y almacenar los valores calculados. De lo contrario, cada vez que el experimento accede a la hoja de cálculo para patrones de datos, los valores deben ser recalculados.

**Nota:** Si los valores de datos pueden cambiar, en su lugar utilice la función de macro BUFFER . Esto crea una copia dinámica del rango de datos, donde los valores copiados cambiarán si los correspondientes valores de entrada cambian.

**Nota:** Si una función de usuario se construye a partir de una definición de función utilizando la función de macro CONSTANT, la parte de la definición de función encerrada en la función de macro CONSTANT se considera una constante. No serán necesarias todas las variables de entrada para aplicar la función de usuario.

## Ejemplos

<code>TEMP = CONSTANT(4.3)</code>
Crea una nueva columna de nombre TEMP que contiene el valor 4.3.
<code>TEMP = CONSTANT(V1)</code>
Crea una nueva columna de nombre TEMP, donde cada valor es una copia estática del contenido de la columna V1.
<code>TEMP = CONSTANT (V1:V3)</code>
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son copias del contenido de la columna V1, los valores de la columna VX son copias del contenido de la columna V2, y los valores de la columna VY son copias del contenido de la columna V3.
<code>TEMP = CONSTANT(V1[10:20])</code>
Crea una nueva columna de nombre TEMP, donde las primeras 11 celdas contienen copias de los valores en las filas 10-20 de la columna V1. Las otras celdas de TEMP están vacías.
<code>TEMP = CONSTANT(V1[50:99]:V2)</code>
Crea dos nuevas columnas de nombre TEMP y VX, cada uno con valores en las filas 1-50 (las otras celdas están vacías). Los valores en la columna TEMP son copias de las filas en la columna V1, y los valores en la columna VX son copias de los valores en la columna V2.
<code>TEMP = CONSTANT(EXTRACT(!ISERROR(V1:V3), V1:V3))</code>
Crea tres nuevas columnas denominadas TEMP, VX y VY. Las tres columnas contienen las filas válidas (es decir, filas que no contienen ninguna ??? celda) de columnas V1 - V3. La derivación de la función de macro EXTRACT dentro de CONSTANT impide que las columnas TEMP, VX y VY se vuelvan a calcular si las columnas V1 - V3 cambian, evitando así la intensa actividad de la función de macro EXTRACT.

## Funciones relacionadas

Función	Descripción
BUFFER	Copia el rango de datos de entrada y se actualiza dinámicamente

---

## COS

### Sintaxis

`COS(data [, units_keyword])`

### Parámetros

data

Los valores numéricos para calcular el coseno de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

units\_keyword



Esta palabra clave opcional determina si los valores de entrada y los resultados se interpretan como grados o radianes. Seleccione una de las opciones siguientes:

RADIAN - Realiza los cálculos en radianes (valor predeterminado)

DEGREE - Realiza los cálculos en grados

Si no se especifica este parámetro, el valor predeterminado es radianes. (Para convertir de radianes a grados, divida por PI y multiplíquelo por 180.)

Para obtener más detalles sobre la utilización de las palabras claves en IBM Campaign, consulte “Especificaciones de formato” en la página 9.

Para obtener más detalles sobre la utilización de palabrasclave en IBM PredictiveInsight, consulte “Especificaciones de formato” en la página 27.

## Descripción

COS calcula el coseno de los valores del rango de datos especificado. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene el coseno de números en la columna de entrada correspondiente.

## Ejemplos

<code>TEMP = COS(PI) o TEMP = COS(PI, 0) o TEMP = COS(PI, RADIAN)</code>
Devuelve una nueva columna de nombre TEMP que contiene el valor único de -1.
<code>TEMP = COS(90, 1) o TEMP = COS(90, DEGREE)</code>
Devuelve una nueva columna de nombre TEMP que contiene el valor único de cero.
<code>TEMP = COS(V1) o TEMP = COS(V1, 0) o TEMP = COS(V1, RADIAN)</code>
Crea una nueva columna de nombre TEMP, donde cada valor es el coseno (en radianes) del contenido de la columna V1.
<code>TEMP = COS(V1:V3, 1)</code>
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los cosenos del contenido de la columna V1, los valores de la columna VX son los cosenos del contenido de la columna V2, y los valores de la columna VY son los cosenos del contenido de la columna V3. Todos los valores se expresan en grados.
<code>TEMP = COS(V1[10:20])</code>
Crea una nueva columna de nombre TEMP, donde las primeras 11 celdas contienen los cosenos de los valores en las filas 10-20 de la columna V1 (en radianes). Las otras celdas de TEMP están vacías.
<code>TEMP = COS(V1[1:5]:V2)</code>
Crea dos nuevas columnas denominadas TEMP y VX, con valores en las filas 1-5 (las otras celdas están vacías). Los valores en la columnaTEMP son los cosenos de las correspondientes filas de la columna V1, y los valores en la columna VX son los cosenos de las filas correspondientes de la columna V2. Todos los valores se expresan en radianes.

## Funciones relacionadas

Función	Descripción
ACOS	Calcula el número NOT lógico del contenido del rango de datos especificado arcocoseno del contenido del rango de datos especificado
COSH	Calcula el número NOT lógico del contenido del rango de datos especificado coseno hiperbólico del contenido del rango de datos especificado
SIN	Calcula el número NOT lógico del contenido del rango de datos especificado seno del contenido del rango de datos especificado
TAN	Calcula la tangente del contenido del rango de datos especificado

---

## COSH

### Sintaxis

`COSH(data [, units_keyword])`

### Parámetros

`data`

Los valores numéricos para calcular el coseno hiperbólico de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

`units_keyword`

Esta palabra clave opcional determina si los valores de entrada y los resultados se interpretan como grados o radianes. Seleccione una de las opciones siguientes:

RADIAN - Realiza los cálculos en radianes (valor predeterminado)

DEGREE - Realiza los cálculos en grados

Si no se especifica este parámetro, el valor predeterminado es radianes. (Para convertir de radianes a grados, divida por PI y multiplíquelo por 180.)

Para obtener más detalles sobre la utilización de las palabras claves en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

### Descripción

COSH calcula el coseno hiperbólico de los valores en el rango de datos especificado. Para  $x$  en radianes, el coseno hiperbólico de un número es:

$$\cosh(x) = \frac{e^x + e^{-x}}{2}$$

donde  $e$  es el número natural, 2,7182818. COSH devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene el coseno hiperbólico de números en la correspondiente columna de entrada.

**Nota:** Si el valor  $x$  es demasiado grande, un desbordamiento de error se devuelve. Esto ocurre si  $\cosh(x)$  excede el máximo valor de coma flotante 32-bit.

## Ejemplos

TEMP = COSH(0) o TEMP = COSH(0, 0) o TEMP = COSH(0, RADIAN)
Devuelve una nueva columna de nombre TEMP que contiene el valor uno.
TEMP = COSH(V1)
Crea una nueva columna de nombre TEMP, donde cada valor es el coseno hiperbólico (en radianes) del contenido de la columna V1.
TEMP = COSH(V1:V3, 1) o TEMP = COSH(V1:V3, DEGREE)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los cosenos hiperbólicos del contenido de la columna V1, los valores de la columna VX son los cosenos hiperbólicos del contenido de la columna V2, y los valores de la columna VY son los cosenos hiperbólicos del contenido de la columna V3. Todos los valores se expresan en grados.
TEMP = COSH(V1[10:20])
Crea una nueva columna de nombre TEMP, donde las primeras 11 celdas contienen los cosenos hiperbólicos de los valores en las filas 10-20 de la columna V1 (en radianes). Las otras celdas de TEMP están vacías.
TEMP = COSH(V1[1:5]:V2)
Crea dos nuevas columnas denominadas TEMP y VX, con valores en las filas 1-5 (las otras celdas están vacías). Los valores de la columna TEMP son los cosenos hiperbólicos de las correspondientes filas de la columna V1, y los valores en la columna VX son los cosenos hiperbólicos de las correspondientes filas de la columna V2. Todos los valores se expresan en radianes.

## Funciones relacionadas

Función	Descripción
ACOS	Calcula el número NOT lógico del contenido del rango de datos especificado arcocoseno del contenido del rango de datos especificado
COS	Calcula el coseno del contenido del rango de datos especificado
SINH	Calcula el seno hiperbólico del contenido del rango de datos especificado
TANH	Calcula la tangente hiperbólica del contenido del rango de datos especificado

# COT

## Sintaxis

`COT(data [, units_keyword])`

## Parámetros

`data`

Los valores numéricos para calcular la cotangente de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

`units_keyword`

Esta palabra clave opcional determina si los valores de entrada y los resultados se interpretan como grados o radianes. Seleccione una de las opciones siguientes:

RADIAN - Realiza los cálculos en radianes (valor predeterminado)

DEGREE - Realiza los cálculos en grados

Si no se especifica este parámetro, el valor predeterminado es radianes. (Para convertir de radianes a grados, divida por PI y multiplíquelo por 180.)

Para obtener más detalles sobre la utilización de las palabras claves en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

## Descripción

COT calcula la cotangente de los valores del rango de datos especificado. La cotangente es recíproca a la tangente. COT devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene la cotangente de números en la correspondiente columna de entrada.

**Nota:** Si una celda contiene un valor cuya tangente es cero, el arcocotangente es infinito. En este caso, COT devuelve el mayor número de coma flotante de 32-bit.

## Ejemplos

<code>TEMP = COT(90) o TEMP = COT(90, 0) o TEMP = COT(90, RADIAN)</code>
Devuelve una nueva columna de nombre TEMP que contiene el valor -0.5.
<code>TEMP = COT(0)</code>
Devuelve una nueva columna de nombre TEMP que contiene el valor MAX_FLOAT_32.
<code>TEMP = COT(V1, 1) o TEMP = COT(V1, DEGREE)</code>
Crea una nueva columna de nombre TEMP, donde cada valor es la cotangente de los contenidos (en grados) de la columna V1.

TEMP = COT(V1:V3, 1)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los cotangentes del contenido de la columna V1, los valores de la columna VX son los cotangentes del contenido de la columna V2, y los valores de la columna VY son los cotangentes del contenido de la columna V3. Todos los valores se expresan en grados.
TEMP = COT(V1[10:20])
Crea una nueva columna de nombre TEMP, donde las primeras 11 celdas contienen los cotangentes de los valores en las filas 10-20 de la columna V1 (en radianes). Las otras celdas de TEMP están vacías.
TEMP = COT(V1[1:5]:V2)
Crea dos nuevas columnas denominadas TEMP y VX, con valores en las filas 1-5 (las otras celdas están vacías). Los valores en la columna TEMP son los cotangentes de las filas correspondientes de la columna V1, y los valores en la columna VX son los cotangentes de las filas correspondientes de la columna V2. Todos los valores se expresan en radianes.

## Funciones relacionadas

Función	Descripción
ACOT	Calcula el número NOT lógico del contenido del rango de datos especificado arccotangente del contenido del rango de datos especificado
COS	Calcula el coseno del contenido del rango de datos especificado
SIN	Calcula el seno del contenido del rango de datos especificado
TAN	Calcula la tangente del contenido del rango de datos especificado

## RECuento

### Sintaxis

RECuento(datos)

### Parámetros

data

El rango de celdas para contar el número de celdas en. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

RECuento recuenta el número de valores en el rango de datos especificado. Devuelve una nueva columna que contiene un único valor que representa el número de celdas que contenga valores dentro del rango de datos especificado.

**Nota:** Recuento de una columna en blanco devuelve cero.

## Ejemplos

TEMP = RECuento(AVG(V1:V5))
Crea una nueva columna de nombre TEMP que contiene un valor único de uno (la función AVG devuelve una celda única en el modo predeterminado).
TEMP = RECuento(V1)
Crea una nueva columna de nombre TEMP que contiene un valor único que indica el número de celdas que contiene valores en la columna V1.
TEMP = RECuento(V1:V3)
Crea una nueva columna de nombre TEMP que contiene un valor único que indica el número de celdas que contienen valores en las columnas V1, V2, y V3.
TEMP = RECuento(V1[10:20])
Crea una nueva columna de nombre TEMP que contiene el valor 11 (los rangos son incluidos), ya que todas las celdas contienen valores.
TEMP = RECuento(V1[1:5]:V4)
Crea una nueva columna de nombre TEMP que contiene el valor 20 (5 celdas en cada columna por 4 columnas = 20 celdas), ya que todas las celdas contienen valores.
TEMP = RECuento(V1[1:10])
Crea una nueva columna de nombre TEMP que contiene el valor 3, ya que las filas 1-3 de la columna V1 contienen valores y las filas 4-10 están vacías.

## Funciones relacionadas

Función	Descripción
RECuento_DIFF	Devuelve cada valor exclusivo en la entrada con un recuento del número de veces que el valor ha aparecido
DESPLAZAMIENTO	Devuelve el desplazamiento de cada valor en la columna de entrada de la parte superior de la corriente
SUM o TOTAL	Calcula la suma de un rango de celdas

---

## RECuento\_DIFF



### Sintaxis

COUNT\_DIFF(data)

### Parámetros

data

El rango de celdas para contar los valores exclusivos y la frecuencia de su ocurrencia. Este puede ser un valor constante, una columna, un rango de celdas o

una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

COUNT\_DIFF encuentra el número de valores diferentes en el rango de datos especificado y recuenta el número de veces que aparece cada valor. Devuelve dos nuevas columnas. La primera columna contiene cada valor exclusivo. La segunda columna contiene el número de veces que el valor correspondiente aparece en el rango de datos. COUNT\_DIFF ignora el caso cuando se comparan las cadenas de texto. Por ejemplo, "Aaa" y "aAa" y "AAA" se recuentan como la misma clase.

**Nota:** La función de macro COUNT\_DIFF puede tardar un tiempo considerable para ser calculada cuando los data son grandes. Un "cálculo..." barra de progreso se visualizará hasta que el cálculo se haya completado. Si decide cancelar el cálculo, pulsar la "X" en la barra de progreso y suprimir la definición de función que contiene la función de macro COUNT\_DIFF .

## Ejemplos

TEMP = COUNT_DIFF(COLUMN(1,2,3,1))	Crea dos nuevas columnas denominadas TEMP y VX. TEMP contiene los valores 1, 2 y 3. VX contiene los recuentos 2, 1, y 1.
TEMP = COUNT_DIFF(COLUMN("x", "a", "a", "b"))	Crea dos nuevas columnas denominadas TEMP y VX. TEMP contiene los valores x, a, y b. VX contiene los recuentos 1, 2 y 1.
TEMP = COUNT_DIFF(V1)	Crea dos nuevas columnas de nombre TEMP y VX, donde TEMP contiene todos los valores exclusivos en la columna V1, y VX contiene un recuento para cada fila correspondiente de TEMP.
TEMP = COUNT_DIFF(V1:V3)	Crea dos nuevas columnas de nombre TEMP y VX, donde TEMP contiene todos los valores exclusivos en las columnas V1 - V3, y VX contiene un recuento para cada fila correspondiente de TEMP.
TEMP = COUNT_DIFF(V1[10:20])	Crea dos nuevas columnas de nombre TEMP y VX, donde TEMP contiene todos los valores exclusivos en las filas 10-20 de la columna V1, y VX contiene un recuento para cada fila correspondiente de TEMP.
TEMP = COUNT_DIFF(V1[1:5]:V4)	Crea dos nuevas columnas de nombre TEMP y VX, donde TEMP contiene todos los valores exclusivos en las filas 1-5 de las columnas V1 - V4, y VX contiene un recuento para cada fila correspondiente de TEMP.

## Funciones relacionadas

Función	Descripción
COUNT	Cuenta la cantidad de celdas que contienen valores en el rango de datos especificado

Función	Descripción
HISTOGRAM	Calcula el histograma de un rango de datos especificado con la utilización de límites de agrupaciones proporcionados

## COV



### Sintaxis

COV(data1, data2)

### Parámetros

data1

El primer conjunto de datos. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

data2

El segundo conjunto de datos. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

COV cuenta la covarianza de los dos rangos de datos especificados. <sup>1</sup> Genera como salidas tantas columnas como la mínima dimensión de las de entrada (ambas, en ancho y altura) . COV se calcula de la siguiente manera:

$$COV(x, y) = \sum_{i=1}^n \frac{(x_i - \mu_x)(y_i - \mu_y)}{N}$$

donde  $x$  y  $y$  son columnas que contienen el mismo número de valores  $N$ ,  $\mu_x$  es el promedio de  $x$ ,  $\mu_y$  es el promedio de  $y$ .

### Ejemplos

```
TEMP = COV(COLUMN(1,2), COLUMN(1,3))
```

Crea una nueva columna de nombre TEMP que contiene el valor 0.5.

```
TEMP = COV(V1, V2)
```

Crea una nueva columna de nombre TEMP que contiene la covarianza entre datos en la columna V1 y datos en la columna V2.



```
TEMP = COV(V1:V2, V3:V4)
```

Crea dos nuevas columnas de nombre TEMP y VX. La columna TEMP contendrá la covarianza entre las columnas V1 y V3. La columna VX contendrá la covarianza entre las columnas V2 y V4.

```
TEMP = COV(V1[1:50]:V2,V3[100:200]:V4)
```

Crea dos nuevas columnas de nombre TEMP y VX. La columna TEMP contendrá la covarianza entre las filas 1-50 de la columna V1 y entre las filas 100-200 de la columna V3. La columna VX contendrá la covarianza entre las filas 1-50 de la columna V2 y entre las filas 100-200 de la columna V4.

---

## CURRENT\_DATE

### Sintaxis

```
CURRENT_DATE([format])
```

### Parámetros

format

Una de las palabras clave en la siguiente tabla especificando el formato de fecha de date\_string.

### Descripción

CURRENT\_DATE devuelve la fecha actual en format. La fecha está determinada por el reloj en el servidor IBM . Si no se proporciona la palabra clave format , se utiliza el valor predeterminado de DELIM\_M\_D\_Y .

Para todas las bases de datos recomendados, IBM Campaign intenta ejecutar la FECHA\_ACTUAL macro en la base de datos utilizando una base de datos soportada actual de tiempo llamada SQL (por ejemplo, SYSDATE, GETDATE, FECHA, o HOY). En estos casos, todos los parámetros (incluido el formato de la fecha) de esta función de macro se ignoran y la salida incluye cualquier cosa devuelta por la base de datos (por ejemplo, un componente de tiempo se puede incluir en la salida). Si ocurre esto y desea devolver sólo la fecha o la fecha en un formato diferente, puede escribir su propia macro personalizada utilizando un SQL sin formato o utilizar otros macros IBM . Por ejemplo:

```
DATE_STRING(CURRENT_JULIAN( ), ...)
```

En algunos casos, la FECHA\_ACTUAL() macro se ejecuta en el servidor de IBM Campaign (por ejemplo, si se ejecuta en un archivo sin formato, contra una base de datos no recomendada sin soporte SQL equivalente, o si la expresión de la campaña macro no puede resolverse en la base de datos). En estos casos, todos los parámetros son reconocidas y la salida se devuelve en el formato seleccionado.

**Nota:** Consulte "Formato Válido de Palabras Clave" para obtener más información sobre formatos de fechas disponibles.

Este macro está disponible en IBM Interact.

## Ejemplos

Si la fecha de hoy es 7 de septiembre de 2000, CURRENT\_DATE() devuelve "07.09.2000".

## Funciones relacionadas

Función	Descripción
DATE_FORMAT	Convierte las fechas de un formato a otro.
DATE_JULIAN	Devuelve la fecha juliana de la entrada.
DATE_STRING	Devuelve la cadena de fecha de la fecha juliana.
DATE	Convierte una cadena de fecha a fecha juliana.

---

## CURRENT\_DAY

### Sintaxis

CURRENT\_DAY

### Descripción

CURRENT\_DAY devuelve el día actual del mes como un número entre 1-31. El tiempo está determinado por el reloj del sistema en el servidor de IBM .

Este macro está disponible en IBM Interact.

### Ejemplos

Si la fecha de hoy es 19 de junio, CURRENT\_DAY() devolverá el número 19.

## Funciones relacionadas

Función	Descripción
CURRENT_JULIAN	Devuelve el número Julián para la fecha actual.
CURRENT_MONTH	Devuelve el mes actual como un número.
CURRENT_TIME	Devuelve la hora actual como serie.
CURRENT_WEEKDAY	Devuelve el día actual de la semana como un número.
CURRENT_YEAR	Devuelve el año actual como un número.

---

## CURRENT\_JULIAN

### Sintaxis

CURRENT\_JULIAN()

## Descripción

CURRENT\_JULIAN() devuelve el número de juliano para la fecha actual (el número de días transcurridos desde enero, 1, 0000). Esto es equivalente a la macro DATE(CURRENT\_DATE()).

## Ejemplos

Si la fecha de hoy es 31 de agosto de 2000, CURRENT\_JULIAN() devuelve el número 730729.

## Funciones relacionadas

Función	Descripción
CURRENT_DAY	Devuelve el día actual como un número.
CURRENT_MONTH	Devuelve el mes actual como un número.
CURRENT_TIME	Devuelve la hora actual como serie.
CURRENT_WEEKDAY	Devuelve el día actual de la semana como un número.
CURRENT_YEAR	Devuelve el año actual como un número.

---

## MES\_ACTUAL

### Sintaxis

CURRENT\_MONTH()

### Descripción

CURRENT\_MONTH devuelve el mes actual del año como un número entre 1-12.

Este macro está disponible en IBM Interact.

### Ejemplos

Si la fecha de hoy es 19 de junio, CURRENT\_MONTH() devolverá el número 6.

### Funciones relacionadas

Función	Descripción
CURRENT_DAY	Devuelve el día actual como un número.
CURRENT_JULIAN	Devuelve el número juliano actual.
CURRENT_TIME	Devuelve la hora actual como serie.
CURRENT_WEEKDAY	Devuelve el día actual de la semana como un número.
CURRENT_YEAR	Devuelve el año actual como un número.

---

## TIEMPO\_ACTUAL

### Sintaxis

CURRENT\_TIME()

### Descripción

CURRENT\_TIME devuelve la hora actual como serie. El tiempo está determinado por el reloj del sistema en el servidor de IBM .

## Configuración de la fecha en su aplicación web

Para visualizar correctamente las fechas en su aplicación web en versiones actuales de IBM Campaign, antes debe configurar correctamente el archivo de configuración del servidor de servicios de fondo. Esto es especialmente importante para los parámetros `dDateFormat` y `DateOutputFormatString` para la base de datos que contiene las tablas del sistema. Si no están configurados correctamente, las fechas se visualizarán en forma incorrecta en campaña. Configura estas propiedades utilizando IBM Marketing Platform.

### Para establecer fechas para un idioma específico en su aplicación web

**Nota:** Todos los archivos de referencia se instalan con el instalador de aplicación web a menos que se indique específicamente.

**Importante:** `webapphome` refiere al directorio donde se ha instalado la aplicación web de la campaña. `language_code` refiere a las configuraciones de idiomas que elige para su sistema.

1. Edite el archivo `webapphome/conf/campaign_config.xml` para asegurarse de que `language_code` está presente en la lista por comas separadas en la etiqueta `<supportedLocales>`, como se muestra a continuación:  

```
<supportedLocales>en_US, language_code</supportedLocales>
```
2. En el directorio `webapphome/webapp`, copie la totalidad del árbol del directorio `en_US` en `language_code` (distinción entre mayúsculas y minúsculas).
3. En `webapphome/webapp/WEB-INF/classes/resources`, copie `StaticMessages_en_US.properties` a `StaticMessages_ language_code.properties`. También copie `ErrorMessages_en_US.properties` a `ErrorMessages_ language_code.properties`.
4. Edite el `StaticMessages_ language_code.properties`: busque `DatePattern` y cámbielo para leer `DatePattern=dd/MM/AAAA` (distinción entre mayúsculas y minúsculas).

**Nota:** Este formato lo define Java™. Se pueden encontrar los detalles completos acerca del formato se pueden encontrar enJava documentación para `java.text.SimpleDateFormat` en <http://java.sun.com>. El archivo `StaticMessages.properties` no necesita modificarse.

5. Para WebSphere: vuelva a jar la aplicación web.
6. Para WebLogic: Elimine el módulo de aplicación web actual.
  - a. Añada el nuevo módulo.
  - b. Vuelva a desplegar la aplicación web.
  - c. No es necesario reiniciar el escucha Campaign.

7. Asegúrese de que el valor del idioma del navegador web `language_code` ha empezado como prioridad. Para obtener más detalles, consulte las siguientes secciones, para definir en su navegador un idioma correcto y para establecer en su sistema un idioma específico.

**Nota:** Asegúrese de utilizar un guión, en contraposición con un carácter subrayado, en `language_code`. La configuración de la aplicación web es el único lugar donde se utiliza un guión en lugar de un subrayado.

8. Inicie sesión en Campaign (Campaña). Las fechas se deben visualizar en Campaign (Campaña) en el formato especificado en `StaticMessages_language_code.properties`.

Para obtener información sobre cómo configurar la hora para IBM Campaign, consulte la documentación *IBM Campaign*.

## Ejemplos

Si la hora es 10:54 a.m., `CURRENT_TIME()` devolverá la cadena "10:54:00 AM".

## Funciones relacionadas

Función	Descripción
<code>CURRENT_DAY</code>	Devuelve el día actual como un número.
<code>CURRENT_JULIAN</code>	Devuelve el número juliano actual.
<code>CURRENT_WEEKDAY</code>	Devuelve el día actual de la semana como un número.
<code>CURRENT_YEAR</code>	Devuelve el año actual como un número.

---

## CURRENT\_WEEKDAY

### Sintaxis

`CURRENT_WEEKDAY()`

### Descripción

`CURRENT_WEEKDAY` devuelve el día de la semana actual en forma de número entre 0 y 6. El domingo se representa como 0, el lunes como 1, etc.

Este macro está disponible en IBM Interact.

### Ejemplos

Si hoy es viernes, `CURRENT_WEEKDAY()` devolverá el número 5.

## Funciones relacionadas

Función	Descripción
<code>CURRENT_DAY</code>	Devuelve el día actual como un número.
<code>CURRENT_JULIAN</code>	Devuelve el número juliano actual.
<code>CURRENT_MONTH</code>	Devuelve el mes actual como un número.
<code>CURRENT_TIME</code>	Devuelve la hora actual como serie.

Función	Descripción
CURRENT_YEAR	Devuelve el año actual como un número.

---

## CURRENT\_YEAR

### Sintaxis

CURRENT\_YEAR()

### Descripción

CURRENT\_YEAR devuelve el año actual en forma de número.

Este macro está disponible en IBM Interact.

### Ejemplos

Si el año actual es 2000, CURRENT\_YEAR() devolverá el número: 2000.

### Funciones relacionadas

Función	Descripción
CURRENT_DAY	Devuelve el día actual como un número.
CURRENT_JULIAN	Devuelve el número juliano actual.
CURRENT_MONTH	Devuelve el mes actual como un número.
CURRENT_TIME	Devuelve la hora actual como serie.
CURRENT_WEEKDAY	Devuelve el día actual de la semana como un número.
MONTHOF	Devuelve el mes del año como número.
WEEKDAYOF	Devuelve el día de la semana como número.
YEAROF	Devuelve el año como número.

---

## CV\_FOLDS



### Sintaxis

CV\_FOLDS(num\_folds, data [, class\_data] [seed])

### Parámetros

num\_folds

Es el número de pliegues a crear por validación cruzada. Este valor debe ser un entero positivo mayor que 1. Este valor debe ser menor que 65,536 o que el número de filas en data, que siempre es menor.

data

Las variables de entrada. Este puede ser una columna, un rango de celdas o una expresión que evalúa a cualquiera de los anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

`class_data`

Si se proporciona este rango de datos opcional, la función de macro `CV_FOLDS` creará pliegues mientras mantiene incluso probabilidades de clase. Los contenidos de `class_data` se utilizan como las salidas por cada patrón de entrada correspondiente.

Si `class_data` es una sola columna, `CV_FOLDS` supone que la columna especificada contiene valores para varias clases de salida (es decir, cada valor distinto se considera una clase aparte). Si `class_data` es un rango de datos, cada columna de salida es considerada una clase diferente. (Con un rango de datos, los valores de cada columna serían uno si un patrón pertenece a esa clase, o cero si el patrón no pertenece a esa clase).

Para obtener la definición de formato de `class_data` (al igual que `data`), consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

`seed`

Es un valor de inicio para utilizar como generador de números aleatorios. Este debe ser entero.

## Descripción

`CV_FOLDS` divide uniformemente los datos de entrada en el número especificado de pliegues. Cada pliegue contendrá el mismo número de patrones de entrada. <sup>2</sup> Esto coloca cada fila del rango de datos de entrada en un pliegue al devolver una nueva columna que contenga números de pliegues con valor entre uno y `num_folds`.

Si se proporciona el parámetro opcional `class_data`, la información de clase de salida se utiliza para crear pliegues de validación cruzada de forma que se mantengan las probabilidades de clase de salida. Es decir, dentro de cada pliegue, la probabilidad de cada clase de salida será la misma. <sup>3</sup>

## Ejemplos

```
TEMP = CV_FOLDS(3, V1, 0)
```

Crea una nueva columna denominada `TEMP` que contiene un valor para cada fila de la columna `V1`. La columna `TEMP` contendrá los valores 1, 2 y 3 para los tres pliegues diferentes. No se mantiene ninguna probabilidad de clase. Se utiliza el valor cero como valor de inicio para el generador de números aleatorios.

```
TEMP = CV_FOLDS(100, V1:V15)
```

Crea una nueva columna denominada `TEMP` que contiene un valor por cada fila de la columna más corta en `V1-V15`. La columna `TEMP` contendrá los valores 1 a 100 para los 100 pliegues diferentes. No se mantiene ninguna probabilidad de clase. Se selecciona un valor de inicio aleatorio.

```
TEMP = CV_FOLDS(50, V1:V10, V11)
```

Crea una nueva columna denominada TEMP que contiene un valor para cada fila de la columna más corta en V1-V10. La columna TEMP contendrá los valores 1 a 50 para los 50 pliegues diferentes. La columna V11 contiene las clases de salida. Cada pliegue tendrá las mismas probabilidades de clase de salida. Se selecciona un valor de inicio aleatorio.

```
TEMP = CV_FOLDS(10, V1:V10, V11:V15, 96)
```

Crea una nueva columna denominada TEMP que contiene un valor para cada fila de la columna más corta en V1-V10. La columna TEMP contendrá los valores 1 a 10 para los 10 pliegues diferentes. Cada una de las columnas de salida V11-V15 representa una clase de salida. Cada pliegue tendrá las mismas probabilidades de clase de salida. Se utiliza el valor 96 como valor de inicio para el generador de números aleatorios.

---

## DATALINK



### Sintaxis

```
DATALINK([spreadsheet,] cells)
```

### Parámetros

spreadsheet

Es el nombre de la hoja de cálculo en la sesión de trabajo de IBM PredictiveInsight actual con la que se va a enlazar (por ejemplo, Sheet1). Si no se especifica ningún valor, se utiliza la hoja de cálculos actual.

cells

Son las celdas específicas dentro de spreadsheet con las que enlazar. El parámetro cells especifica la o las columnas y la o las filas con las que enlazar y se puede especificar como cualquiera de las siguientes:

```
Cn Cn:Cm RnRn | CmRn RnRn:RNCM | CmRn:CMRN
```

Los caracteres C y R especifican columna y fila respectivamente. Las variables n, m, N y M son los números de fila y columna.

### Descripción

DATALINK crea un enlace interno con datos en la sesión de trabajo de IBM PredictiveInsight actual. Devuelve el número de columnas especificado en el parámetro cells. Los cambios en la columna de datos de origen se reflejarán en forma automática.

### Ejemplos

```
TEMP = DATALINK(C1)
```

Crea una nueva columna denominada TEMP que contiene los valores en la columna uno de la hoja de cálculo actual.



TEMP = DATALINK(Sheet2, C1:C3)

Crea tres nuevas columnas denominadas TEMP, VX y VY, que contienen los valores en las columnas 1 a 3 de la hoja de cálculo **Sheet2**.

TEMP = DATALINK(Sheet4, C5R10) o TEMP = DATALINK(Sheet4, R10C5)

Crea una nueva columna denominada TEMP que contiene el valor de celda en la quinta columna, décima fila de la hoja de cálculo **Sheet4**.

TEMP = DATALINK(Sheet1, C1R1:C3R500) o TEMP = DATALINK(Sheet1, R1C1:R500C3)

Crea tres nuevas columnas denominadas TEMP, VX y VY, que contienen los valores en las filas 1 a 500 de las columnas 1 a 3 de la hoja de cálculo **Sheet2**.

## Funciones relacionadas

Función	Descripción
DDELINK	Crea un enlace externo a los datos de otra aplicación de Windows

---

## DATE

### Sintaxis

DATE(date\_string [, format])

### Parámetros

date\_string

Un texto que representa una fecha válida.

format

Una de las palabras clave en la tabla bajo "Formato válido de palabras clave", que especifica el formato de fecha de date\_string.

### Descripción

DATE convierte una cadena de fecha en una fecha juliana (el número de días transcurridos desde el 1° de enero de 0000). Prácticamente cualquier formato de fecha es soportado a través de la palabra clave opcional format, que especifica cómo se representa la fecha. Si no se proporciona la palabra clave format, se utiliza el formato predeterminado DELIM\_M\_D\_Y.

Los formatos de fecha son de anchura fija (por ejemplo, la fecha 28 de febrero de 1970 se representa como 02281970 en el formato MMDDAAA), o delimitadas (por ejemplo, 28 de febrero de 1970, 2-28-1970 o 02/28/1970). Todos los ejemplos anteriores son variantes del formato DELIM\_M\_D\_YY.

En formatos delimitados, los delimitadores son la barra inclinada (/), el guión(-), el espacio ( ), la coma (,) o los dos puntos (:); el año se puede representar con 2 o 4 dígitos; y el mes se puede escribir completo (por ejemplo, febrero), abreviado (por ejemplo, feb) o en forma numérica (por ejemplo, 2 o 02).

Para todos los años especificados con dos dígitos:

- Los años de dos dígitos menores al umbral del año 2000 (el valor predeterminado es 20, pero puede ser establecido por el usuario) se considera que están después del 2000.
- Los años de dos dígitos mayores o iguales al umbral se considera que están en el 1900.

Este macro está disponible en IBM Interact.

## Ejemplos

DATE("8/31/2000") devuelve el número 730729.

Formato de palabras clave válidas

Palabra clave	Descripción	Ejemplo(s)
MM	Mes de dos dígitos	01, 02, 03, ..., 12
MMDD	Mes de dos dígitos y día de dos dígitos	31 de marzo es 0331
MMDDYY	Mes de dos dígitos, día de dos dígitos y año de dos dígitos	31 de marzo de 1970 es 033170
MMDDYYYY	Mes de dos dígitos, día de dos dígitos y año de cuatro dígitos	31 de marzo de 1970 es 03311970
DELIM_M_D	Cualquier mes delimitado seguido por el día	31 de marzo, 3/31 o 03-31
DELIM_M_D_Y	Cualquier mes, día y año delimitado	31 de marzo de 1970 o 3/31/70
DELIM_Y_M	Cualquier año delimitado seguido por el mes	Marzo, 70; 3-70 o 3/1970
DELIM_Y_M_D	Cualquier año, mes y día delimitado	1970 mar 31 o 70/3/31
YYMMM	Año de 2 dígitos y mes de 3 letras	70MAR
YYMMDD	Año de 2 dígitos, mes de 3 letras y día de 2 dígitos	70MAR31
YY	Año de 2 dígitos	70
YYMM	Año de 2 dígitos y mes de 2 dígitos	7003
YYMMDD	Año de 2 dígitos, mes de 2 dígitos y día de 2 dígitos	700331
YYYYMMM	Año de 4 dígitos y mes de 3 letras	1970MAR
YYYYMMDD	Año de 4 dígitos, mes de 3 letras y días de 2 dígitos	1970MAR31
YYYY	Año de 4 dígitos	1970
YYYYMM	Año de 4 dígitos y mes de 2 dígitos	197003
YYYYMMDD	Año de 4 dígitos, mes de 2 dígitos y día de 2 dígitos	19700331

DELIM_M_Y	Cualquier mes delimitado seguido por el año	3-70, 3/70, mar 70, marzo 1970
DELIM_D_M	Cualquier día delimitado seguido por el mes	31-3, 31/3, 31 de marzo
DELIM_D_M_Y	Cualquier día, mes y año delimitado	31-MAR-70, 31/3/1970, 31 03 70
DD	Día de 2 dígitos	31
DDMMM	Día de 2 dígitos y mes de 3 letras	31MAR
DDMMYY	Día de 2 dígitos, mes de 3 letras y año de 2 dígitos	31MAR70
DDMMYYYY	Día de 2 dígitos, mes de 3 letras y año de 4 dígitos	31MAR1970
DDMM	Día de 2 dígitos y mes de 2 dígitos	3103
DDMMYY	Día de 2 dígitos, mes de 2 dígitos y año de 2 dígitos	310370
DDMMYYYY	Día de 2 dígitos, mes de 2 dígitos y año de 4 dígitos	31031970
MMYY	Mes de 2 dígitos y año de 2 dígitos	0370
MMYYYY	Mes de 2 dígitos y año de 4 dígitos	031970
MMM	Mes de 3 letras	MAR
MMMDD	Mes de 3 letras y día de 2 dígitos	MAR31
MMMDDYY	Mes de 3 letras, día de 2 dígitos y año de 2 dígitos	MAR3170
MMMDDYYYY	Mes de 3 letras, día de 2 dígitos y año de 4 dígitos	MAR311970
MMYY	Mes de 3 letras y año de 2 dígitos	MAR70
MMYYYY	Mes de 3 letras y año de 4 dígitos	MAR1970
MONTH	Mes del año	Enero, febrero, marzo, etc. o ene, feb, mar, etc.
WEEKDAY	Día de la semana	Domingo, lunes, martes, etc. (domingo = 0)
WKD	Día de la semana abreviado	Dom, lun, mar, etc. (Dom = 0)

## Funciones relacionadas

Función	Descripción
DATE_FORMAT	Convierte las fechas de un formato a otro.
DATE_JULIAN	Devuelve la fecha juliana de la entrada.

Función	Descripción
DATE_STRING	Devuelve la cadena de fecha de la fecha juliana.
CURRENT_DATE	Devuelve la fecha actual en un formato especificado.

---

## DATE\_FORMAT

### Sintaxis

DATE\_FORMAT(date\_string, input\_format, output\_format)

### Parámetros

date\_string

Un texto que representa una fecha válida.

input\_format

Una de las palabras clave en la tabla siguiente que especifica el formato de fecha de date\_string.

output\_format

Una de las palabras clave en la tabla siguiente especifica el formato de fecha de salida deseado.

### Descripción

DATE\_FORMAT() transforma una fecha de input\_format a otro formato output\_format.

Si la fecha tiene una anchura fija, se debe establecer en uno de los siguientes valores:

- DDMMYY[YY]
- DDMMMYY[YY]
- MMDDYY[YY]
- MMMDDYY[YY]
- YY[YY]MMDD
- YY[YY]MMMDD

MM es un mes en 2 dígitos y MMM es el mes abreviado en 3 caracteres.

Si la fecha se delimita la fecha (se puede utilizar cualquier delimitador usando ESPACIO, GUIÓN, BARRA INCLINADA), se debe establecer en uno de estos valores:

- DELIM\_D\_M\_Y
- DELIM\_M\_D\_Y
- DELIM\_Y\_M\_D

Este macro está disponible en IBM Interact.

## Ejemplos

DATE\_FORMAT("012171", MMDDYY, MMDDYYYY) devuelve la cadena "01211971".

**Nota:** Consulte "DATE" en la página 77 para obtener más información sobre formatos de fecha válidos.

Funciones relacionadas

Función	Descripción
DATE	Convierte una cadena de fecha en una fecha juliana
DATE_JULIAN	Devuelve la fecha juliana de la entrada.
DATE_STRING	Devuelve la cadena de fecha de la fecha juliana.

---

## DATE\_JULIAN

### Sintaxis

DATE\_JULIAN(year, month, day)

### Parámetros

year

Número de año válido de 2 dígitos o de 4 dígitos.

month

Número de mes válido entre 1 y 12.

day

Número de día válido entre 1 y 31.

### Descripción

DATE\_JULIAN devuelve la fecha juliana de la entrada especificada. La fecha juliana es el número de días transcurridos desde el 1º de enero de 0000.

### Ejemplos

DATE\_JULIAN (2000,08,31) devuelve el número 730729.

### Funciones relacionadas

Función	Descripción
DATE	Convierte una cadena de fecha en una fecha juliana
DATE_FORMAT	Convierte las fechas de un formato a otro.
DATE_STRING	Devuelve la cadena de fecha de la fecha juliana.

---

## DATE\_STRING

### Sintaxis

```
DATE_STRING(julian_date [, 'output_format'[, max_length]])  
DATE_STRING(julian_date [, 'format_string'[, max_length]])
```

### Parámetros

julian\_date

Es un número que representa una fecha juliana, el número de días transcurridos desde el 1° de enero de 0000.

output\_format

Cadena, formato de fecha válido.

max\_length

format\_string

Una serie de formato que incluye en forma opcional cualquier combinación de las siguientes codificaciones de formato:

Código	Descripción
%a	Nombre del día de la semana abreviado
%A	Nombre completo del día de la semana
%b	Nombre abreviado del mes
%B	Nombre completo del mes
%c	Representación de la fecha y hora adecuada para el entorno local
%d	Días del mes (01 - 31)
%H	La hora en formato de 24 horas (00 - 23)
%I	La hora en formato de 12 horas (01 - 12)
%j	Día del año (001 - 366)
%m	Mes (01 - 12)
%M	Minuto (00 - 59)
%p	Actual indicador de AM/PM de entorno local para reloj de 12 horas
%S	Segundo (00 - 59)
%U	Semana del año, con el domingo como primer día de la semana (00 - 51)
%w	Día de la semana (0 - 6; el domingo es 0)
%W	Semana del año, con el lunes como el primer día de la semana (00 - 51)
%x	Representación de la fecha para el entorno local actual
%X	Representación de la hora en el entorno local actual

Código	Descripción
%y	Año de 2 dígitos (00 - 99)
%Y	Año de 4 dígitos
%z, %Z	Nombre o abreviatura de la zona horaria; no hay salida si se desconoce la zona horaria
%%	Signo de porcentaje

## Descripción

DATE\_STRING devuelve la cadena de fecha de la fecha juliana. Si no se proporciona output\_format, se utilizará la palabra clave predeterminada DELIM\_M\_D\_Y.

## Ejemplos

DATE\_STRING(730729) devuelve la cadena "08/31/00".

**Nota:** Consulte "DATE" en la página 77 para obtener más información sobre formatos de fecha válidos.

## Funciones relacionadas

Función	Descripción
DATE	Convierte una cadena de fecha en una fecha juliana
DATE_JULIAN	Devuelve la fecha juliana de la entrada.
DATE_FORMAT	Convierte las fechas de un formato a otro.

---

## DAY\_BETWEEN

### Sintaxis

DAY\_BETWEEN(from\_date\_string, to\_date\_string [, input\_format])

### Parámetros

from\_date\_string

Un texto que representa una fecha válida desde la que contar el número de días transcurridos.

to\_date\_string

Un texto que representa una fecha válida hasta la que se cuentan los días. Esta fecha debe estar en el mismo formato que from\_date\_string.

input\_format

Una de las palabras clave en la tabla siguiente, que especifica el formato de fecha de from\_date\_string y to\_date\_string.

## Descripción

DAY\_BETWEEN devuelve el número de días entre `from_date_string` y `to_date_string`. Si se proporciona `input_format`, se utilizará la palabra clave predeterminada `DELIM_M_D_Y`.

## Ejemplos

DAY\_BETWEEN("08/25/00","08/31/00") devuelve el número 6.

**Nota:** Consulte "DATE" en la página 77 para obtener más información sobre formatos de fecha válidos.

## Funciones relacionadas

Función	Descripción
DAY_FROMNOW	Devuelve el número de días desde la fecha actual a una fecha especificada.
DAY_INTERVAL	Devuelve el número de días entre dos fechas especificadas.

---

## DAY\_FROMNOW

### Sintaxis

DAY\_FROMNOW(`to_year`, `to_month`, `to_day`)

### Parámetros

`to_year`

Número de año válido de 2 dígitos o de 4 dígitos.

`to_month`

Número de mes válido entre 1 y 12.

`to_day`

Número de día válido entre 1 y 31.

### Descripción

DAY\_FROMNOW devuelve el número de días entre el día actual y la fecha especificada por `to_year/to_month/to_day`.

**Nota:** Si la fecha especificada es del pasado, el valor devuelto será negativo.

### Ejemplos

Si hoy es el 31 de agosto de 2000, DAY\_FROMNOW(2000,12,31) devuelve el número 122.



## Funciones relacionadas

Función	Descripción
DAY_BETWEEN	Devuelve el número de días entre dos cadenas de fecha especificadas.
DAY_INTERVAL	Devuelve el número de días entre dos fechas especificadas.

---

## DAY\_INTERVAL

### Sintaxis

DAY\_INTERVAL(from\_year, from\_month, from\_day, to\_year, to\_month, to\_day)

### Parámetros

from\_year

Número de año válido de 2 dígitos o de 4 dígitos.

from\_month

Número de mes válido entre 1 y 12.

from\_day

Número de día válido entre 1 y 31.

to\_year

Número de año válido de 2 dígitos o de 4 dígitos.

to\_month

Número de mes válido entre 1 y 12.

to\_day

Número de día válido entre 1 y 31.

### Descripción

DAY\_INTERVAL devuelve el número de días entre la fecha de origen especificada (from\_year/from\_month/from\_day) y la fecha de finalización especificada (to\_year/to\_month/to\_day).

### Ejemplos

DAY\_INTERVAL(2000,8,31,2000,12,31) devuelve el número 122.

## Funciones relacionadas

Función	Descripción
DAY_BETWEEN	Devuelve el número de días entre dos cadenas de fecha especificadas.
DAY_FROMNOW	Devuelve el número de días desde la fecha actual a una fecha especificada.

---

## DAYOF

### Sintaxis

DAYOF(date\_string [, input\_format])

### Parámetros

date\_string

Un texto que representa una fecha válida.

input\_format

Una de las palabras claves en la tabla siguiente que especifica el formato de fecha de date\_string.

### Descripción

DAYOF devuelve el día del mes como un número para la fecha representada por el date\_string. Si no se proporciona el input\_format, se utilizará la palabra clave predeterminada DELIM\_M\_D\_Y.

### Ejemplos

DAYOF("08/31/00") devuelve el número 31.

**Nota:** Consulte "DATE" en la página 77 para obtener más información sobre formatos de fecha válidos.

---

## DDELINK



### Sintaxis

DDELINK(service, topic, items)

### Parámetros

service

Es el nombre del servicio (por ejemplo, excel) con el que crear un enlace DDE.

topic

Es el tema dentro deservice con el que se desea enlazar. Para la mayoría de las aplicaciones, el tema es un nombre de archivo. Escriba la ruta completa y el nombre de archivo del tema deseado (por ejemplo, c:\stock\prices\05jan.xls).

items

Son los elementos en topic con los que enlazar. La sintaxis de los elementos depende del servicio seleccionado. Por ejemplo, en Excel, R1C1:R10C20 selecciona las filas 1 a 10 y las columnas 1 a 20.

## Descripción

DDELINK crea un enlace de intercambio de datos dinámico (DDE) con datos en una aplicación de Windows externa. Devuelve el número de columnas especificado en el parámetro items. Los cambios en la aplicación de origen se verán reflejados en forma automática en la hoja de cálculo de IBM PredictiveInsight.

## Ejemplos

TEMP = DDELINK(Excel, c:\excel\data.xls, C1:C2)
Crea dos columnas nuevas denominadas TEMP y VX, que contienen los valores en las columnas 1 y 2 de la hoja de cálculo c:\excel\data.xls.
TEMP = DDELINK(Excel, c:\excel\data.xls, R1:R10)
Crea tantas columnas como sean necesarias para acomodar las filas 1 a 10 de la hoja de cálculo c:\excel\data.xls.
TEMP = DDELINK(Excel, c:\excel\data.xls, R1C1:R100C3)
Crea tres nuevas columnas denominadas TEMP, VX y VY que contienen las filas 1 a 100 de la hoja de cálculo c:\excel\data.xls.
TEMP = DDELINK(123W, c:\lotsuite\sample.wk4, A:A1..A:C8)
Crea tres nuevas columnas denominadas TEMP, VX y VY contienen las filas 1 a 8 de las columnas A a C de la hoja de cálculo A en el archivo de Lotus sample.wk4.

## Funciones relacionadas

Función	Descripción
DATALINK	Crea un enlace interno a los datos en una hoja de cálculo de IBM PredictiveInsight

---

## DECIMATE



### Sintaxis

DECIMATE(column, max\_value)

### Parámetros

column

La columna de valores a los que se desea aplicar la función. Todos los valores dentro de esta columna deben ser enteros positivos menores que max\_value.

max\_value

El número de columnas a devolver. Este debe ser un entero positivo mayor o igual al máximo valor en column.

## Descripción

DECIMATE convierte un valor entero positivo en un patrón binario de max\_value columnas de longitud. Si el valor es *n*, la columna *n* contiene un uno; todas las otras columnas contienen ceros. La función de macro devuelve max\_value columnas.

**Nota:** DECIMATE es la opuesta a la función de macro MAXINDEX.

## Ejemplos

TEMP = DECIMATE(COLUMN(1,2,3), 3)
Crea columnas nuevas denominadas TEMP, VX y VY con una fila para cada fila correspondiente de entrada. La primera fila contiene 1 0 0, la segunda contiene 0 1 0 y la tercera contiene 0 0 1.
TEMP = DECIMATE(COLUMN(1,1,2), 3)
Crea columnas nuevas denominadas TEMP, VX y VY con una fila para cada fila correspondiente de entrada. La primera fila contiene 1 0 0, la segunda contiene 1 0 0 y la tercera contiene 0 1 0.
TEMP = DECIMATE(V1, 10)
Crea diez columnas nuevas con una fila por cada fila correspondientes de entrada. Cada fila contiene un solo uno en la columna representando el correspondiente valor de entrada. Todas las otras columnas contienen ceros.

## Funciones relacionadas

Función	Descripción
MAX	Calcula el número máximo de un rango de celdas
MAXINDEX	Devuelve los valores de las columnas de entrada rezagadas por un número especificado de pasos de tiempo
MIN	Calcula el número mínimo de un rango de celdas

---

## DELAY



### Sintaxis

DELAY(delay, data)

## Parámetros

delay

Es el número de pasos de tiempo a retardar. Este valor debe ser un entero positivo.

data

Son los valores del retardo. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

DELAY devuelve valores en el rango de datos de entrada, retardados por el número de pasos de tiempo especificados. Considera cada columna de entrada como una serie de datos en el tiempo y devuelve una nueva columna para cada columna de entrada. Cada columna nueva contiene los valores de retardo de pasos de tiempo (retardado por el número de pasos de tiempo delay) de los números de la correspondiente columna de entrada.

**Nota:** La función de macro DELAY devuelve una columna con los valores de tal manera que la celda  $VY[x] = data[x + delay]$ .

**Nota:** Esta función es útil para crear patrones desde datos de series de tiempo. Para crear varios retardos, utilice la función de macro SLIDE\_WINDOW.

## Ejemplos

```
TEMP = DELAY(1, COLUMN(1,2,3,4))
```

Crea una nueva columna denominada TEMP que contiene los valores 2, 3 y 4 en las celdas 1 a 3 respectivamente.

```
TEMP = DELAY(2, V1)
```

Crea una nueva columna denominada TEMP, donde cada valor es el contenido de la columna V1 retardado por dos pasos de tiempo.

```
TEMP = DELAY(10, V1:V3)
```

Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los contenidos de la columna V1 retardados por diez pasos de tiempo, los valores de la columna VX son los contenidos de la columna V2 retardados por diez pasos de tiempo y los valores de la columna VY son los contenidos de la columna V3 retardados por diez pasos de tiempo.

```
TEMP = DELAY(1, V1[10:20])
```

Crea una nueva columna denominada TEMP, donde las primeras 10 celdas contienen los valores en la columna V1 retardados por un paso de tiempo (es decir, las filas 11 a 20 de la columna V1). Las otras celdas de TEMP están vacías.

```
TEMP = DELAY(2, V1[1:5]:V2)
```

Crea dos columnas nuevas denominadas TEMP y VX, cada una con valores en las filas 1 a 3 (las otras celdas están vacías). Los valores de la columna TEMP son las filas correspondientes de la columna V1 retardadas por dos pasos de tiempo y los valores en la columna VX son las filas correspondientes de la columna V2 retardadas por dos pasos de tiempo (es decir, las filas 3 a 5 de las columnas V1 y V2).

## Funciones relacionadas

Función	Descripción
LAG	Devuelve los valores de las columnas de entrada rezagadas por un número especificado de pasos de tiempo
SLIDE_WINDOW	Crea un patrón desde una ventana especificada y lo desliza hacia abajo para crear el siguiente patrón

---

## DERIVATIVE



### Sintaxis

DERIVATIVE(data [, divisor])

### Parámetros

data

Son los valores numéricos para calcularles el derivado. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

divisor

Es el valor entre el que se divide cada valor de data. Esta puede ser un valor constante o una expresión que evalúe a una constante.

### Descripción

DERIVATIVE calcula el derivado de los valores en una serie de tiempo. Cada valor es la diferencia entre el valor actual y el valor del próximo paso de tiempo. Si se proporciona el valor para divisor, cada valor se divide entre el valor especificado. DERIVATIVE devuelve una columna nueva por cada columna de entrada, cada una de las cuales contiene los derivados de los valores en la correspondiente columna de entrada.

**Nota:** La longitud de la columna devuelta será uno menos que la longitud de la columna de datos de origen (data).

### Ejemplos

TEMP = DERIVATIVE(5)
Crea una nueva columna denominada TEMP que contiene todas las celdas en blanco (se necesita al menos dos valores de celda para generar un resultado).
TEMP = DERIVATIVE(COLUMN(1, 2, 5))
Crea una nueva columna denominada TEMP, que contiene los valores 1 y 3.

TEMP = DERIVATIVE(V1)
Crea una nueva columna denominada TEMP, donde cada valor es el derivado de los contenidos de la columna V1.
TEMP = DERIVATIVE(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los derivados de los contenidos de la columna V1, los valores de la columna VX son los derivados de los contenidos de la columna V2 y los valores de la columna VY son los derivados de los contenidos de la columna V3.
TEMP = DERIVATIVE(V1[10:20])
Crea una nueva columna denominada TEMP, donde las celdas en las filas 1 a 20 contienen los derivados de las correspondientes filas de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = DERIVATIVE(V1[1:5]:V2)
Crea dos columnas nuevas denominadas TEMP y VX, cada una con valores en las filas 1 a 5 (las otras celdas están vacías). Los valores en la columna TEMP son los derivados de las correspondientes filas de la columna V1 y los valores de la columna VX son los derivados de las correspondientes filas de la columna V2.

## Funciones relacionadas

Función	Descripción
INTEGRAL	Calcula la integral de los valores del rango de datos especificado

---

## DISTANCE

### Sintaxis

DISTANCE(lat1, long1, lat2, long2[, UNIT\_OF\_MEASURE] [, PRECISION])

### Parámetros

lat1

La latitud del primer punto, como valor decimal.

long1

La longitud del primer punto, como valor decimal.

lat2

La latitud del segundo punto, como valor decimal.

long2

La longitud del segundo punto, como valor decimal.

UNIT\_OF\_MEASURE

Parámetro opcional que indica la unidad de medida de la distancia devuelta. Los valores son MILES o KILOMETERS. Si omite este parámetro, el valor predeterminado es MILES.

PRECISION

Parámetro opcional que indica el nivel de precisión que sigue al separador decimal de la distancia devuelta. Si especifica un valor de precisión, la distancia devuelta se trunca en el número de posiciones decimales que especifique. El valor máximo es 5. Si omite este valor, el número de posiciones decimales no se trunca.

## Descripción

DISTANCE calcula la distancia entre dos puntos. Se espera que la latitud y la longitud estén en unidades decimales.

## Ejemplos

DISTANCE (18.529747,73.839798,18.533511,73.8777995,MILES,2) devuelve el valor de 2,50 millas.

---

## DISTINCT



### Sintaxis

DISTINCT(data)

### Parámetros

data

Es el rango de datos.

### Descripción

DISTINCT encuentra los valores exclusivos en el rango de datos especificado. Devuelve esta lista de valores en una sola columna. DISTINCT no distingue mayúsculas y minúsculas cuando compara cadenas de texto. Por ejemplo, \042Aaa\042 y \042aAa\042 y \042AAA\042 se calculan como el mismo valor.

La función de macro DISTINCT puede tomar un tiempo largo para calcular cuando los datos son grandes. Se visualizará una barra de progreso \042Computing...\042 hasta que se complete el cálculo. Si decide cancelar el cálculo, pulse \042X\042 en la barra de progreso y suprima la definición de función que contiene la función de macro DISTINCT.

---

## DIV

### Sintaxis

data DIV divisor data / divisor



## Parámetros

data

Los valores numéricos por los que dividir. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúe a cualquiera de los anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

divisor

El o los valores por los que se dividen los valores en el rango de datos especificado. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en data2 debe ser igual al número de columnas en data1, a menos que data2 sea una constante. Para obtener la definición de formato del divisor (al igual que de data), consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

DIV divide el rango de datos especificado por el valor del divisor. Devuelve una columna nueva por cada columna de entrada, cada una de las cuales contiene la columna correspondiente en data1 dividida por la correspondiente columna de data2 (es decir, la primera columna de data1 se divide por la primera columna de data, la segunda columna con la segunda columna, etc.).

Si data2 es una constante, cada valor en data1 se divide por ese valor. Si data2 contiene una o más columnas, los cálculos se realizan fila por fila entre una columna de data1 y una columna de data2. La primera fila de data1 se divide por el valor de primera fila de data2, la segunda fila con la segunda fila y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** Utilizar una columna con el mismo número  $x$  en cada fila como divisor es lo mismo que utilizar la constante  $x$  como divisor.

**Nota:** El operador DIV se puede abreviar con una barra inclinada (/).

Este macro está disponible en IBM Interact.

## Ejemplos

TEMP = 8 DIV 4 o TEMP = 8/4
Crea una nueva columna denominada TEMP que contiene el valor dos.
TEMP = V1/8
Crea una nueva columna denominada TEMP, donde cada valor es el contenido de la columna V1 dividido por ocho.

TEMP =V1:V3/2
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los contenidos de la columna V1 divididos entre dos, los valores de la columna VX son los contenidos de la columna V2 divididos entre dos y los valores de la columna VY son los contenidos de la columna V3 divididos entre dos.
TEMP = V1/V1
Crea una nueva columna denominada TEMP que contiene todo unos (ya que todo número dividido por sí mismo da como resultado uno).
TEMP = V1/V2
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 dividido por el correspondiente valor de fila de la columna V2.
TEMP = V1:V3/V4:V6
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores de V1 divididos por el correspondiente valor de fila de la columna V4. La columna VX contiene la división de la columna V2 entre V5. La columna VY contiene la división de la columna V3 entre V6.
TEMP = V1[10:20] / V2 o TEMP = V1[10:20] / V2[1:11]
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen el resultado de dividir los valores de las filas 10 a 20 de la columna V1 entre los valores de las filas 1 a 11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
MOD	Calcula el módulo del contenido del rango de datos especificado
MULT	Multiplica el contenido de dos rangos de datos
POW	Calcula un valor base elevado a las potencias exponenciales especificadas

## EQ

### Sintaxis

data1 EQ data2 data1 == data2 (data1 = data2)

### Parámetros

data1

El rango de celdas para comparar. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

data2

El/los número(s) con los que comparar todos los valores de la columna especificada. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en data2 debe ser igual al número de columnas en data1, a menos que data2 sea una constante. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

EQ compara los dos rangos de datos especificados y devuelve un uno si los valores son iguales o un cero si no lo son. Devuelve un nueva columna para cada columna de entrada, cada una de las cuales contiene la correspondiente columna en data1 comparada con la columna correspondiente de data2 (es decir que la primera columna de data1 se compara con la primera columna de data, la segunda columna con la segunda columna, y a sí sucesivamente).

Si data2 es una constante, cada valor en data se compara con ese valor. Si data2 es una columna, los cálculos se realizan fila por fila. Los valores en data1 se comparan con el valor de la primera fila de data2, la segunda fila con la segunda fila y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

Cuando se comparan cadenas, no se distingue entre mayúsculas y minúsculas (es decir que "Yes", "YES", "yes" y "yeS" se consideran iguales).

**Nota:** El operador EQ se puede abreviar con dos signos de igual (==). También se puede utilizar un signo igual entre paréntesis (=) para la función de macro EQ (sin paréntesis, el signo igual se interpreta como operador de asignaciones).

Este macro está disponible en IBM Interact.

## Ejemplos

TEMP = 3 EQ 4 o TEMP = 3==4 o TEMP = (3=4)
Crea una nueva columna denominada TEMP que contiene el valor cero (ya que tres no es igual a cuatro).
TEMP = "No" == "NO"
Crea una nueva columna denominada TEMP que contiene el valor uno (las comparaciones de cadenas no distinguen entre mayúsculas y minúsculas).
TEMP = V1 == 8
Crea una nueva columna denominada TEMP, donde cada valor es uno si el valor de fila correspondiente de la columna V1 es igual al número ocho, de lo contrario es cero.
TEMP = V1==V1
Crea una nueva columna denominada TEMP que contiene solo unos (dado que todo número es igual a sí mismo).
TEMP = V1==V2
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 comparado con el valor de fila correspondiente de la columna V2.

TEMP = V1:V3 == V4:V6
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores en V1 comparados con los valores de fila correspondientes de la columna V4. La columna VX compara las columnas V2 y V5. La columna VY compara las columnas V3 y V6.
TEMP = V1[10:20] == V2 o TEMP = V1[10:20] == V2[1:11]
Crea una nueva columna denominada TEMP, donde las once primeras celdas contienen los resultados de comparar los valores en las filas 10 a 20 de la columna V1 con las filas 1 a 11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
EQ	Devuelve TRUE si un rango de datos es igual a otro
GE	Devuelve TRUE si un rango de datos es mayor o igual a otro
GT	Devuelve TRUE si un rango de datos es mayor que otro
LE	Devuelve TRUE si un rango de datos es menor o igual a otro
LT	Devuelve TRUE si un rango de datos es menor que otro
NE	Devuelve TRUE si un rango de datos no es igual a otro

## EXP

### Sintaxis

EXP(data)

### Parámetros

data

Es el valor numérico utilizado como exponente de un número natural,  $e$ . Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúe a cualquiera de los anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

EXP eleva el número natural,  $e$ , a cada uno de los valores en el rango de datos especificado (es decir, calcula  $e^x$ ). La constante  $e$  equivale a 2.7182818. EXP devuelve una columna nueva por cada columna de entrada, cada una de las cuales contiene el resultado  $e^x$  por cada valor  $x$  en la o las correspondientes columnas de entrada. EXP es la inversa de la función de macro LN.

**Nota:** Si el valor  $x$  es demasiado grande o demasiado pequeño, se devuelve un error por desbordamiento. Esto sucede si  $e^x$  supera el máximo o mínimo de 32 bits de valor de coma flotante.

## Ejemplos

TEMP = EXP(2)	Crea una nueva columna denominada TEMP que contiene el valor 7.39.
TEMP = EXP(V1)	Crea una nueva columna denominada TEMP, donde cada valor es el resultado de elevar $e$ a los contenidos de la columna V1.
TEMP = EXP(V1:V3)	Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son los resultados de elevar $e$ a la columna V1, los valores de la columna VX son los resultados de elevar $e$ a los contenidos de la columna V2 y los valores de la columna VY son los resultados de elevar $e$ a los contenidos de la columna V3.
TEMP = EXP(V1[10:20])	Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los resultados de elevar $e$ a los valores de las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = EXP(V1[1:5]:V2)	Crea dos columnas nuevas denominadas TEMP y VX, cada una con valores en las filas 1 a 5 (las otras celdas están vacías). Los valores de la columna TEMP son los resultados de elevar $e$ a los correspondientes valores de fila de la columna V1 y los valores de la columna VX son los resultados de elevar $e$ a los correspondientes valores de fila de la columna V2.

## Funciones relacionadas

Función	Descripción
LN o LOG	Calcula el logaritmo natural del contenido del rango de datos especificado
LOG2	Calcula el logaritmo en base 2 del contenido del rango de datos especificado
LOG10	Calcula el logaritmo en base 10 del contenido del rango de datos especificado
POW	Potencia exponencial

---

## EXTERNALCALLOUT



### Sintaxis

EXTERNALCALLOUT( *calloutName*, *arg1*, ...)

### Parámetros

*calloutName*

Es el nombre de la llamada que ha creado utilizando la API ExternalCallout. Este nombre debe coincidir con el nombre de la categoría de llamada externa que ha creado en IBM Marketing Platform.

arg1

Es un argumento necesario para la llamada, si es necesario.

## Descripción

EXTERNALCALLOUT le permite llamar a una aplicación externa para añadir datos al diagrama de flujo interactivo. EXTERNALCALLOUT puede devolver lo que sea que la llamada haya sido creada para hacer. Debe escribir esta llamada en Java utilizando la API ExternalCallout. Para obtener más detalles, consulte la Guía del desarrollador de *IBM Interact*.

## Ejemplos

```
EXTERNALCALLOUT(getStockPrice, UNCA)
```

Llama a la llamada `getStockPrice` pasando el nombre de la bolsa, `UNCA`, como argumento. Esta llamada definida por el usuario devuelve el precio de la bolsa como lo define la llamada.

---

## EXTRACT



### Sintaxis

```
EXTRACT(predicate_col, data)
```

### Parámetros

`predicate_col`

Una columna de valores booleanos o una expresión que se evalúa en una sola columna de valores booleanos. Los valores booleanos se interpretan como cero o distinto de cero. Esta columna debe contener al menos tantas filas como el rango de datos del que se extraen los datos. De lo contrario, `predicate_col` será un límite para el número de filas procesadas por la función de macro EXTRACT (consulte "Descripción" más abajo).

`data`

Los valores a extraer. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúe a cualquiera de los anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

EXTRACT devuelve las filas en el rango de datos especificado que tienen un valor de uno en la fila correspondiente de la columna predicado. Esta función de macro reduce los datos al "arrojar afuera" todas las filas con un valor correspondiente en `predicate_col` de cero. EXTRACT devuelve una columna nueva por cada columna de

entrada, cada una de las cuales contiene los valores en la correspondiente columna de entrada para los que el valor correspondiente en `predicate_col` es uno. Las filas de datos extraídas ocuparán las primeras  $n$  celdas de las columnas de salida donde  $n$  es el número de unos en `predicate_col`.

Debido a que EXTRACT opera fila por fila, produce un resultado por cada fila hasta el último valor de la columna más corta (es decir, la columna más corta de `predicate_col` y las columnas en el rango de datos `data`). Todos los valores distintos de cero en `predicate_col` se evalúan como uno.

**Nota:** Por lo general, deseará crear una columna predicado utilizando una de las funciones de macro de comparación (por ejemplo, `==`, `>`, `<`, `ISEVEN`, `ISODD`, etc.). Puede entonces extraer las filas de interés de un rango de datos especificado utilizando la función de macro EXTRACT. Esto puede ser útil para "limpiar de" puntos de datos errados (por ejemplo, cuando un valor determinado supera el máximo o mínimo valor para una variable de datos). También se puede utilizar para consolidar todos los ejemplos de una determinada clase (por ejemplo, si la columna V3 contiene unos y ceros para una de las clases de salida, utilice `V4=EXTRACT(V3, V1:V2)` para extraer las entradas V1 y V2). Debido a que EXTRACT condensa todas las filas extraídas como un bloque de datos (es decir, rellena el rango de datos `VX[1:n]:VY`), donde  $n$  es el número de filas extraídas, es una función útil para copiar un rango de celdas desde la ubicación actual de fila a las filas 1-  $n$  de la hoja de cálculo.

## Ejemplos

<code>TEMP = EXTRACT(1, V1)</code>	Crea una nueva columna denominada TEMP que contiene una copia de la columna V1.
<code>TEMP = EXTRACT(1, V1[50:100]:V2)</code>	Crea dos columnas nuevas denominadas TEMP y VX con valores en las primeras 51 celdas. Los valores en la columna TEMP son las celdas 50 a 100 de la columna V1 y los valores en la columna VX son las celdas 50 a 100 de la columna V2.
<code>TEMP = EXTRACT(V3, V1:V2)</code>	Crea dos nuevas columnas denominadas TEMP y VX. Para cada fila con el valor uno en la columna V3, el valor correspondiente en las columnas V1 y V2 se extraen en las columnas TEMP y VX, respectivamente.
<code>TEMP = EXTRACT(V1&gt;V2, V1)</code>	Crea una nueva columna denominada TEMP que contiene todos los valores de la columna V1 mayores que los valores correspondientes de la columna V2.
<code>TEMP = EXTRACT(V3[10:20], V1[10:20]:V2)</code>	Crea dos nuevas columnas denominadas TEMP y VX. Para las filas 10 a 20 con un valor de uno en la columna V3, la fila correspondiente en las columnas V1 y V2 se extraen en las columnas TEMP y VX, respectivamente.

## Funciones relacionadas

Función	Descripción
IF	Comienza una sentencia condicional if-then-else de la sentencia
SELECT	Devuelve la(s) columna(s) especificada(s) de un rango de datos

Función	Descripción
SUBSAMPLE	Reduce los datos al devolver cada enésimo valor de fila
SUBSTITUTE	Reemplaza los valores de una columna con un valor especificado en una tabla de conversión

## FACTORIAL

### Sintaxis

FACTORIAL(data)

### Parámetros

data

Los valores enteros para calcularles el factorial. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúe a cualquiera de los anteriores, pero debe ser mayor o igual a cero. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

FACTORIAL calcula el factorial de los valores en el rango de datos especificado. Todas las entradas deben ser enteros mayores o iguales a cero. El factorial de un entero menor o igual a uno es uno. Para enteros  $x \geq 2$ , el factorial  $x! = x(x-1)(x-2)\dots(x - (x-1))$ . FACTORIAL devuelve una columna nueva por cada columna de entrada, cada una de las cuales contiene el factorial de números de la correspondiente columna de entrada.

**Nota:** Cualquier valor mayor que 34 producirá ??? (error de desbordamiento de coma flotante).

### Ejemplos

TEMP = FACTORIAL(3)
Crea una nueva columna denominada TEMP que contiene el valor 6.
TEMP = FACTORIAL(-2)
Genera un error 333, que indica que el argumento debe ser mayor o igual a 0.
TEMP = FACTORIAL(V1)
Crea una nueva columna denominada TEMP, donde cada valor es el factorial de los contenidos de la columna V1.
TEMP = FACTORIAL(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los factoriales de los contenidos de la columna V1, los valores de la columna VX son los factoriales de los contenidos en la columna V2 y los valores de la columna VY son los factoriales de los contenidos de la columna V3.



```
TEMP = FACTORIAL(V1[10:20])
```

Crea una nueva columna denominada TEMP, donde las primeras once celdas contienen los factoriales de los valores de las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.

```
TEMP = FACTORIAL(V1[50:99]:V2)
```

Crea dos columnas nuevas denominadas TEMP y VX, cada una con los valores en las filas 1 a 50 (las otras celdas están vacías). Los valores en la columna TEMP son los factoriales de los valores de las filas en la columna V1 y los valores de la columna VX son los factoriales de los valores en la columna V2.

---

## FLOOR

### Sintaxis

```
FLOOR(data)
```

### Parámetros

data

Los valores numéricos para calcularles el múltiplo inferior. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

FLOOR calcula el múltiplo inferior de los valores en el rango de datos especificado. El múltiplo inferior de un número es el mayor entero menor que el número. FLOOR devuelve una columna nueva por cada columna de entrada, cada una de las cuales contiene el múltiplo inferior de números en la correspondiente columna de entrada.

**Nota:** Es lo mismo que la función de macro INT.

### Ejemplos

```
TEMP = FLOOR(4.3)
```

Crea una nueva columna denominada TEMP que contiene el valor 4.

```
TEMP = FLOOR(2.9)
```

Crea una nueva columna denominada TEMP que contiene el valor -3.

```
TEMP = FLOOR(V1)
```

Crea una nueva columna denominada TEMP, donde cada valor es el múltiplo inferior de los contenidos de la columna V1.

```
TEMP = FLOOR(V1:V3)
```

Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los múltiplos inferiores de los contenidos de la columna V1, los valores de la columna VX son los múltiplos inferiores de los contenidos de la columna V2 y los valores de la columna VY son los múltiplos inferiores de los contenidos en la columna V3.

TEMP = FLOOR(V1[10:20])

Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los múltiplos inferiores de los valores en las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.

TEMP = FLOOR(V1[50:99]:V2)

Crea dos nuevas columnas de nombre TEMP y VX, cada uno con valores en las filas 1-50 (las otras celdas están vacías). Los valores de la columna TEMP son los múltiplos inferiores de los valores de las filas de la columna V1 y los valores de la columna VX son los múltiplos inferiores de los valores en la columna V2.

## Funciones relacionadas

Función	Descripción
CEILING	Calcula el ceiling de cada valor del rango de datos especificado
FRACTION	Devuelve la parte fraccionaria de cada valor del rango de datos especificado
TRUNCATE	Devuelve la parte no fraccionaria de cada valor del rango de datos especificado

---

## FORMAT

### Sintaxis

Format tiene dos formatos, uno para tipos de datos numéricos y uno para tipos de datos de texto o caracteres.

Para tipos de datos numéricos:

FORMAT(colName, width [, precision [, format\_type [, alignment [, padding]]]])

Para tipos de datos de texto o caracteres:

FORMAT(colName, width [, alignment])

### Parámetros

colName

La macro examina colName y determina su tipo de datos, después impone las reglas adecuadas para los siguientes parámetros según corresponda.

width

El ancho debe ser lo suficientemente grande para retener el resultado completo, de lo contrario se truncará el resultado. Los valores aceptables son entre 1 y 29 si colName es numérico, de lo contrario entre 1 y 255.

precision

La precisión es el número de dígitos después del separador decimal. Los valores aceptables son entre 0 y 15. Si es cero, el resultado es entero. El valor de precisión predeterminado es 2.

`format_type`

Las palabras claves válidas para `format_type` son:

PERIOD	El punto (.) se utiliza como símbolo decimal. No se utiliza separador de miles. Éste es el valor predeterminado.
COMMA	La coma (,) se utiliza como símbolo decimal. No se utiliza separador de miles.
PERIOD_COMMA	El punto como símbolo decimal y la coma como separador de miles.
COMMA_PERIOD	La coma como símbolo decimal y el punto como separador de miles.

`alignment`

Las palabras clave de alineación válidas son `LEFT` y `RIGHT`. El valor predeterminado para tipos de dato numérico es `RIGHT` y para tipos de datos de texto o caracteres es `LEFT`.

`padding`

Las palabras clave para el relleno válidas son `SPACE` y `ZERO`. El valor predeterminado es `SPACE`. Se ignora `ZERO` (y en su lugar se utiliza `SPACE`) si la alineación es `LEFT`.

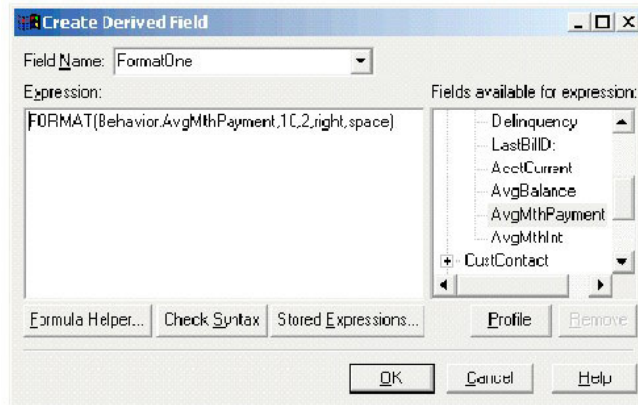
Tenga en cuenta que las cadenas numéricas que se retienen dentro de un tipo de dato de texto a caracteres se tratan como texto o caracteres. También tenga en cuenta que el formato numérico tiene varias palabras claves opcionales, cada una con un valor predeterminado. Sin embargo, para alterar temporalmente el valor predeterminado de una palabra clave opcional segunda o posterior, debe codificar los valores predeterminados para las palabras claves opcionales anteriores (de hecho, se convierten en necesarias). Por ejemplo: para alterar temporalmente la alineación a `LEFT` debe codificar: `FORMAT(myNumCol, 10, 2, PERIOD, LEFT)`.

## Descripción

`FORMAT` convierte datos numéricos a una cadena formada de varias opciones de formato para controlar y definir la cadena de salida. Esto será especialmente útil para crear archivos de instantáneas con formatos específicos para el envío de correos.

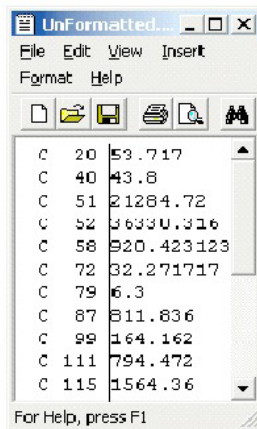
## Ejemplos

El siguiente ejemplo define un campo derivado de FORMAT.



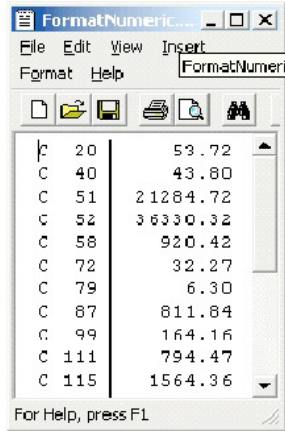
Los siguientes ejemplos muestran el mismo campo, AvgMthPayment, en tres formatos.

Sin formato:

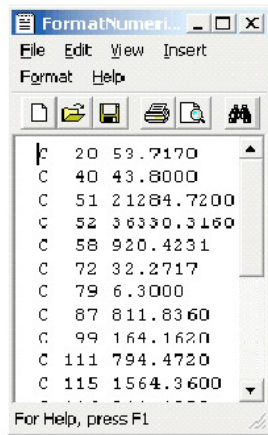


C		
20	53.717	
40	43.8	
51	21284.72	
52	36330.316	
58	920.423123	
72	32.271717	
79	6.3	
87	811.836	
99	164.162	
111	794.472	
115	1564.36	

Formateado utilizando FORMAT(Behavior.AvgMthPayment,10,2,right,space):



Formateado utilizando `FORMAT(Behavior.AvgMthPayment,10,4)`:



## FRACTION

### Sintaxis

`FRACTION(data)`

### Parámetros

`data`

Los valores numéricos para calcularles la fracción. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

`FRACTION` calcula la parte decimal de los valores en el rango de datos especificado. Devuelve una columna nueva por cada columna de entrada, cada una de las cuales contiene la parte decimal de los números de la correspondiente columna de entrada.

**Nota:** Las funciones de macroFRACTION y TRUNCATE son complementarias, ya que su suma da como resultado los valores originales.

## Ejemplos

TEMP = FRACTION(4.3)
Crea una nueva columna denominada TEMP que contiene el valor 0.3.
TEMP = FRACTION(2.9)
Crea una nueva columna denominada TEMP que contiene el valor -0.9.
TEMP = FRACTION(V1)
Crea una nueva columna denominada TEMP, donde cada valor es la parte fraccionaria del contenido de la columna V1.
TEMP = FRACTION(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son la parte decimal de los contenidos de la columna V1, los valores de la columna VX son la parte decimal de los contenidos de la columna V2 y los valores de la columna VY son la parte decimal de los contenidos de la columna V3.
TEMP = FRACTION(V1[10:20])
Crea una nueva columna denominada TEMP, donde las primeras once celdas contienen la parte decimal de los valores de las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = FRACTION(V1[50:99]:V2)
Crea dos columnas nuevas denominadas TEMP y VX, cada una con los valores en las filas 1 a 50 (las otras celdas están vacías). Los valores de la columna TEMP son la parte decimal de las filas de la columna V1 y los valores de la columna VX son la parte decimal de los valores de la columna V2.

## Funciones relacionadas

Función	Descripción
CEILING	Calcula el ceiling de cada valor del rango de datos especificado
FLOOR	Calcula el floor de cada valor del rango de datos especificado
TRUNCATE	Devuelve la parte no fraccionaria de cada valor del rango de datos especificado

---

## GAUSS



### Sintaxis

GAUSS(data [, mean, std])

### Parámetros

data1

El rango de celdas para calcular el gaussiano de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

mean

El promedio del gaussiano. Si no se proporciona este parámetro, el valor predeterminado es cero. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en mean debe ser igual al número de columnas en data, a menos que mean sea una constante o una sola columna. Para obtener la definición de formato de mean, (al igual que la definición de data) consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

std

La desviación estándar del gaussiano. Si no se proporciona este parámetro, el valor predeterminado es uno. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en std debe ser igual al número de columnas en data, a menos que std sea una constante o una sola columna. Para obtener la definición de formato de std, (al igual que la definición de data) consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Sintaxis

GAUSS calcula el valor gaussiano de los números en el rango de datos especificado. Devuelve una columna nueva por cada columna entrada, cada una de las cuales contiene el valor gaussiano de la correspondiente entrada. GAUSS se calcula como se indica a continuación:

$$\frac{1}{\sqrt{2\pi}} e^{-\frac{(x - \mu)^2}{2\sigma^2}}$$

Los parámetros promedio y desviación estándar se tratan como se indica a continuación:

- Si no se proporciona ni el promedio ni la desviación estándar, se utilizan cero y uno respectivamente.
- Si mean y std son constantes, se utiliza estos valores para especificar el gaussiano para todos los valores en data.
- Si mean y std son una sola columna cada uno, se utilizan los valores de fila correspondientes para cada fila de data.
- Si mean y std son rangos de columnas (ambos deben tener el mismo número de columnas que data), cada celda en data utiliza su par individual de celdas correspondientes en mean y std.

## Ejemplos

<code>TEMP = GAUSS(0)</code> o <code>TEMP = GAUSS(0, 0, 1)</code>
Crea una nueva columna denominada TEMP que contiene el valor 0.4.
<code>TEMP = GAUSS(V1)</code>
Crea una nueva columna denominada TEMP, donde cada valor es el gaussiano de la correspondiente fila de la columna V1, utilizando un gaussiano de promedio cero y de varianza unidad.
<code>TEMP = GAUSS(V1:V3)</code>
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los gaussianos de la columna V1, los valores de la columna VX son los gaussianos de la columna V2 y los valores de la columna VY son los gaussianos de la columna V3. El gaussiano es de promedio cero y de varianza unidad.
<code>TEMP = GAUSS(V1[1:50]:V3)</code>
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son los gaussianos de las filas 1 a 50 de la columna V1, los valores de la columna VX son los gaussianos de las filas 1 a 50 de la columna V2 y los valores de las filas de la columna VY son los gaussianos de la columna V3. El gaussiano es de promedio cero y de varianza unidad.
<code>TEMP = GAUSS(V1, 0, 3.5)</code>
Crea una nueva columna denominada TEMP, donde cada valor es el gaussiano de la correspondiente fila de la columna V1. El gaussiano tiene un promedio de 0 y una desviación estándar de 3.5.
<code>TEMP = GAUSS(V1, V2, V3)</code>
Crea una nueva columna denominada TEMP, donde cada valor es el gaussiano de la columna V1, utilizando el correspondiente valor de fila de la columna V2 como promedio y el correspondiente valor de fila de la columna V3 como desviación estándar.
<code>TEMP = GAUSS(V1:V2, V3:V4, V5:V6)</code>
Crea dos columnas nuevas denominadas TEMP y VX. La columna TEMP contiene los gaussianos de los valores en V1 y utiliza las correspondientes filas de la columna V3 como el promedio y las correspondientes filas de la columna V5 como la desviación estándar. La columna VX contiene los gaussianos de los valores en V2 y utiliza las correspondientes filas de la columna V4 como el promedio y las correspondientes filas de la columna V6 como la desviación estándar.

## Funciones relacionadas

<code>RANDOM_GAUSS</code>	Devuelve el número especificado de valores aleatorios de una distribución gaussiana
---------------------------	---

---

## GAUSS\_AREA



### Sintaxis

`GAUSS_AREA(data [, mean, std])`



## Parámetros

data1

El rango de celdas para calcular el área bajo el gaussiano. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de mean, (al igual que la definición de data) consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

mean

El promedio del gaussiano. Si no se proporciona este parámetro, el valor predeterminado es cero. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a cualquiera de los anteriores. El número de columnas en mean debe ser igual al número de columnas en data, a menos que mean sea una constante o una sola columna. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

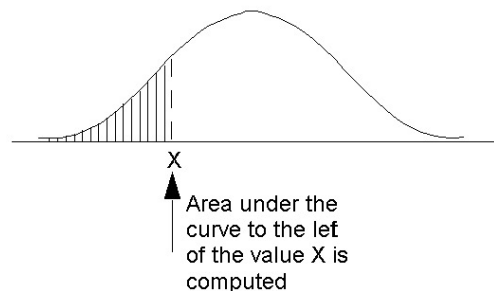
std

La desviación estándar del gaussiano. Si no se proporciona este parámetro, el valor predeterminado es uno. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en std debe ser igual al número de columnas en data, a menos que std sea una constante o una sola columna. Para obtener la definición de formato de std, (al igual que data), consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

GAUSS\_AREA calcula el área bajo el gaussiano (desde  $-\infty$  hasta el valor de datos especificado X) de los números en el rango de datos especificado(consulte la siguiente figura). Devuelve una columna nueva por cada columna de entrada, cada una de las cuales contiene el área bajo el gaussiano de la correspondiente entrada.

Area Under a Gaussian Computed by GAUSS\_AREA



Los parámetros promedio y desviación estándar se tratan como se indica a continuación:

- Si no se proporciona ni el promedio ni la desviación estándar, se utilizan cero y uno respectivamente.

- Si mean y std son constantes, se utiliza estos valores para especificar el gaussiano para todos los valores en data.
- Si mean y std son una sola columna cada uno, se utilizan los valores de fila correspondientes para cada fila de data.
- Si mean y std son rangos de columnas (ambos deben tener el mismo número de columnas que data), cada celda en data utiliza su par individual de celdas correspondientes en mean y std.

**Nota:** En el último caso, cuando mean y std son rangos de columnas, la longitud de cada columna determina cuantas filas estarán presentes en la columna de salida correspondiente. Si una columna de mean o std es una sola celda, ese valor será utilizado para todos los valores de filas de data. Si mean o std contienen varias filas, se calculan las correspondientes filas. Las filas de data para las que no hay valores correspondientes en mean y std no se calculan.

## Ejemplos

<p>TEMP = GAUSS_AREA(0) or TEMP = GAUSS_AREA(0, 0, 1)</p> <p>Crea una nueva columna denominada TEMP que contiene el valor 0.5.</p>
<p>TEMP = GAUSS_AREA(V1)</p> <p>Crea una nueva columna denominada TEMP, donde cada valor es el área bajo el gaussiano de la correspondiente fila de la columna V1, utilizando un gaussiano de promedio cero y de varianza unidad.</p>
<p>TEMP = GAUSS_AREA(V1:V3)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son las áreas bajo los gaussianos de la columna V1, los valores de la columna VX son las áreas bajo los gaussianos de la columna V2 y los valores de la columna VY son las áreas bajo los gaussianos de la columna V3. El gaussiano es de promedio cero y de varianza unidad.</p>
<p>TEMP = GAUSS_AREA(V1[1:50]:V3)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son las áreas bajo los gaussianos de las filas 1 a 50 de la columna V1, los valores de la columna VX son las áreas bajo los gaussianos de las filas 1 a 50 de la columna V2 y los valores de las filas de la columna VY son las áreas bajo los gaussianos de la columna V3. El gaussiano es de promedio cero y de varianza unidad.</p>
<p>TEMP = GAUSS_AREA(V1, 0, 3.5)</p> <p>Crea una nueva columna denominada TEMP, donde cada valor es el área bajo del gaussiano de la correspondiente fila de la columna V1. El gaussiano tiene un promedio de 0 y una desviación estándar de 3.5.</p>
<p>TEMP = GAUSS_AREA(V1, V2, V3)</p> <p>Crea una nueva columna denominada TEMP, donde cada valor es el área bajo el gaussiano de las columna V1, utilizando el valor de fila correspondiente de la columna V2 como el promedio y el valor de fila correspondiente de la columna V3 como desviación estándar.</p>
<p>TEMP = GAUSS_AREA(V1:V2, V3:V4, V5:V6)</p> <p>Crea dos nuevas columnas denominadas TEMP y VX. La columna TEMP contiene las áreas bajo los gaussianos de los valores en V1 utilizando las filas correspondientes de las columna V3 como el promedio y las filas correspondientes de la columna V5 como desviación estándar. La columna VX contiene las áreas bajo los gaussianos de los valores de V2 utilizando las correspondientes filas de la columna V4 como promedio y las correspondientes filas de la columna V6 como desviación estándar.</p>

## Sintaxis

data1 GE data2 data1 >= data2

## Parámetros

data1

El rango de celdas numéricas a comparar. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

data2

El/los número(s) con los que comparar todos los valores de la columna especificada. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en data2 debe ser igual al número de columnas en data1, a menos que data2 sea una constante. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

GE compara los dos rangos de datos especificados y devuelve un uno si los valores del primer conjunto de datos son mayores o iguales a los valores del segundo conjunto de datos o un cero en caso contrario. Devuelve una columna nueva por cada columna de entrada, cada una de las cuales contiene la columna correspondiente en data1 comparada con la correspondiente columna de data2 (es decir, la primera columna de data1 se compara con la primera columna de data, la segunda columna con la segunda columna, etc.).

Si data2 es una constante, cada valor en data se compara con ese valor. Si data2 es una columna, los cálculos se realizan fila por fila. Los valores en data1 se comparan con el valor de la primera fila de data2, la segunda fila con la segunda fila y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** El operador GE puede ser abreviado con un signo mayor que seguido de un signo igual (>=).

Este macro está disponible en IBM Interact.

## Ejemplos

```
TEMP = 9 GE 4 o TEMP = 9 >= 4
```

Crea una nueva columna denominada TEMP que contiene el valor uno (ya que nueve es mayor que cuatro).

TEMP = V1 >= 8
Crea una nueva columna denominada TEMP, donde cada valor es uno si el valor de fila correspondiente de la columna V1 es mayor o igual al número ocho, de lo contrario es cero.
TEMP = V1:V3 >= 2
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los contenidos de la columna V1 comparados con el valor dos, los valores de la columna VX son los contenidos de la columna V2 comparados con el valor dos y los valores de la columna VY son los contenidos de la columna V3 comparados con el valor dos.
TEMP = V1 >= V1
Crea una nueva columna denominada TEMP que contiene todos unos (ya que todos los números son iguales a sí mismos).
TEMP = V1 >= V2
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 comparado con el valor de fila correspondiente de la columna V2.
TEMP = V1:V3 >= V4:V6
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores en V1 comparados con los valores de fila correspondientes de la columna V4. La columna VX compara las columnas V2 y V5. La columna VY compara las columnas V3 y V6.
TEMP = V1[10:20] >= V2 o TEMP = V1[10:20] >= V2[1:11]
Crea una nueva columna denominada TEMP, donde las primeras once celdas contienen el resultado de comparar los valores de las filas 10 a 20 de la columna V1 con los valores de las filas 1 a 11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

NE	Devuelve TRUE si un rango de datos no es igual a otro
----	---

## GRID



### Sintaxis

GRID(col1 [, col2]...)

### Parámetros

col1

La primera columna desde la que se produce la cuadrícula. Puede ser una constante, una columna, un rango de celdas de una sola columna o cualquier expresión que evalúa a una de las anteriores.

col2

Una o más columnas adicionales para utilizar en la producción de la cuadrícula. Puede ser una constante, una columna, un rango de celdas de una sola columna o cualquier expresión que evalúa a una de las anteriores.

## Descripción

GRID genera una cuadrícula de todas las combinaciones posibles de valores que utilizan celdas en cada una de las columnas de entrada. Devuelve una columna nueva por cada columna de entrada. Si alguna de las entradas es una constante, cada valor en las otras columnas de entrada se emparejan con ese valor constante.

Utilizando la notación V1[1] para la primera celda de la columna V1, V1[2] para la segunda celda, etc., la expresión GRID(V1, V2) produciría:

```
V1[1] V2[1] V1[1] V2[2] V1[1] V2[3] : : V1[2] V2[1] V1[2] V2[2] V1[2] V2[3]
: : : :
```

Todas las celdas de la columna V1 son emparejados con cada celda de la columna V2. GRID se comporta de forma similar para más de dos columnas de entrada. El número de filas generadas es igual al producto de la longitud de las columnas de entrada.

**Nota:** La longitud concatenada de todos los argumentos de entrada no puede superar  $(2^{16} - 1)/16$ .

## Ejemplos

TEMP = GRID(1,2)
Crea dos columnas nuevas denominadas TEMP y VX, que contienen los valores 1 y 2, respectivamente.
TEMP = GRID(COLUMN(1, 2), COLUMN(3, 4)) Crea dos columnas nuevas denominadas TEMP y VX. Las filas en estas dos columnas son: 1 3 1 4 2 3 2 4
TEMP = GRID(V1)
Crea una nueva columna denominada TEMP que contiene una copia de valores en la columna V1.
TEMP = GRID(V1, 3)
Crea dos nuevas columnas denominadas TEMP y VX. La columnaTEMP es una copia de la columna V1 y la columna VX contiene el valor 3 para cada fila de la columna V1.
TEMP = GRID(V1, V2)
Crea dos columnas nuevas denominadas TEMP y VX, donde cada fila representa una de las posibles combinaciones de celdas (vea descripción).
TEMP = GRID(V1, V3, V7)
Crea tres nuevas columnas denominadas TEMP, VX y VY, donde cada fila representa una de las posibles combinaciones de celdas (vea descripción).

## Funciones relacionadas

Función	Descripción
SLIDE_WINDOW	Crea un patrón desde una ventana especificada y lo desliza hacia abajo para crear el siguiente patrón

# GROUPBY

## Sintaxis

GROUPBY(group\_field, keyword, rolled\_field [,output\_field])

## Parámetros

- group\_field  
Especifica la variable sobre la que se agrupan los registros (es decir, todos los mismos valores de la variable especificada se agrupan juntos).
- palabra clave  
Especifica el resumen de la función de acumulación a realizar en el rolled-field.
- rolled\_field  
Especifica la variable a resumir o acumular.
- output\_field  
Identifica una variable alternativa a devolver para una sola fila de un grupo y solo puede utilizarse con las palabras clave MinOf, MaxOf y MedianOf.

## Descripción

GROUPBY resume varias filas de datos dentro de un grupos. La salida de esta función es una sola columna. La salida es el resultado de la operación especificada por la keyword en el rolled\_field a través del grupo homogéneo especificado por el group\_field. Si hay más de una respuesta que satisfaga una condición especificada, se devuelve la primera encontrada.

Si no se proporciona el output\_field opcional, la salida es el resultado de la operación en rolled\_field. Si se proporciona output\_field, el resultado es el output\_field de la fila dentro del grupo.

Si hay varias filas dentro de un grupo que satisfacen la condición especificada (Por ejemplo, hay varios enlaces para el valor máximo), se devuelve el output-field asociado con la primera fila que satisface la condición.

**Nota:** Para trabajar con la agrupación de varias columnas, puede especificar una lista de nombres de campo, separados por comas, dentro de un par de corchetes "llaves" "{}" y utilizarlo como el primer parámetro en la llamada macro GROUPBY macro call.

Las palabras claves soportadas son las siguientes (no distingue entre mayúsculas y minúsculas):

Palabra clave	¿Cadena? Si/No	Descripción
CountOf	Yes	Devuelve el número de registros en cada grupo(rolled_field puede ser numérico o cadena; el valor devuelto es el mismo independientemente del valor de rolled_field).

MinOf	Yes	Devuelve el valor mínimo de rolled_field en cada grupo(rolled_field puede ser numérico o cadena; si rolled_field es una cadena, se devuelve el valor más cercano al comienzo del abecedario en orden alfabético).
MaxOf	Yes	Devuelve el valor máximo de rolled_field en cada grupo(rolled_field puede ser numérico o cadena; si rolled_field es una cadena, se devuelve el valor más cercano al final del abecedario en orden alfabético).
DiffOf	Yes	Devuelve el número de valores distintos de rolled_field en cada grupo (rolled_field puede ser numérico o cadena).
AvgOf	No	Devuelve el valor promedio de rolled_field en cada grupo(rolled_field debe ser numérico).
ModeOf	Yes	Devuelve el valor modal (es decir, el valor más repetido) de rolled_field en cada grupo (rolled_field puede ser numérico o cadena).
MedianOf	Yes	Devuelve el valor de la mediana (es decir, el valor del medio ordenado por rolled_field) de rolled_field en cada grupo (rolled_field puede ser numérico o cadena; si rolled_field es una cadena, los valores se ordenan alfabéticamente).
OrderOf	Yes	Devuelve el orden de rolled_field en cada grupo (rolled_field debe ser numérico). Si varios registros tienen el mismo valor, todos ellos reciben el mismo valor.
SumOf	No	Devuelve la suma de rolled_field en cada grupo (rolled_field debe ser numérico).
StdevOf	No	Devuelve la desviación estándar de rolled_field en cada grupo(rolled_field debe ser numérico).

IndexOf	Yes	Devuelve el índice en base 1 (ordenado por rolled_field) de cada registro (rolled_field puede ser numérico o cadena). El orden de clasificación es ascendente.  Nota: Para campos numéricos, el orden de clasificación de RankOf e IndexOf puede ser descendente con colocar un signo de menos (-) delante del campo de clasificación.
RankOf	Yes	Devuelve la categoría en base 1 (ordenada por rolled_field) en que se encuentra cada registro (rolled_field puede ser numérico o cadena). El orden de clasificación es ascendente.  Nota: Para campos numéricos, el orden de clasificación de RankOf e IndexOf puede ser descendente con colocar un signo de menos (-) delante del campo de clasificación.

## Ejemplos

GROUPBY (Household_ID, SumOf, Account_Balance)
Calcula la suma de todos los saldos de cuenta por unidad familiar.
GROUPBY (Cust_ID, MinOf, Date(Account_Open_Date), Acc_Num)
Devuelve el número de cuenta de la primera cuenta abierta por el cliente.

---

## GROUPBY\_WHERE

### Sintaxis

GROUPBY\_WHERE(group\_field, keyword, rolled\_field, where\_value [,output\_field])

### Parámetros

- group\_field  
Especifica la variable sobre la que se agrupan los registros (es decir, todos los mismos valores de la variable especificada se agrupan juntos).
- palabra clave  
Especifica la función de agrupación de resumen a realizar.
- rolled\_field



Especifica la variable a resumir o acumular.

- `where_value`

Una expresión que evalúa en un valor de uno o cero que especifica que filas se incluirán en la operación de acumulación.

- `output_field`

Identifica una variable alternativa a devolver para una sola fila de un grupo y se puede utilizar únicamente con las palabras clave `MinOf`, `MaxOf` y `MedianOf`

## Descripción

`GROUPBY_WHERE` resume filas de datos específicas dentro de un grupo. La salida de esta función es una sola columna. La salida es el resultado de la operación especificada por `keyword` en el `rolled_field` sobre el grupo homogéneo especificado por el `group_field`, filtrado por el `where_value`. Solo las filas con un `where_value` de uno se incluyen en el cálculo.

Si no se proporciona el `output_field` opcional, el resultado es el resultado de la operación en `rolled_field`. Si se proporciona `output_field`, el resultado es el `output_field` de la fila dentro del grupo.

**Nota:** Consulte "GROUPBY" en la página 114 para obtener más información sobre valores válidos para `keyword`.

## Ejemplos

```
GROUPBY_WHERE (Household_ID, SumOf, Account_Balance, Account_Balance>0)
```

Calcula la suma de todas las cuentas con saldos positivos para cada unidad familiar.

```
GROUPBY_WHERE (Cust_ID, AvgOf, Purchase_Amt, Date(Current_Date) -  
Date(Purchase_Date)<90)
```

Calcula el número NOT lógico del contenido del rango de datos especificado promedio de importe de compras por cada cliente para compras en los últimos 90 días.

---

## GT

### Sintaxis

```
data1 GT data2 data1 > data2
```

### Parámetros

`data1`

El rango de celdas numéricas a comparar. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúe a cualquiera de los anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

`data2`

Los números contra los que comparar todos los valores de la columna especificada. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en

data2 debe ser igual al número de columnas en data1, a menos que data2 sea una constante. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

GT compara los dos rangos de datos especificados y devuelve un uno si los valores del primer conjunto de datos son mayores que los valores del segundo conjunto de datos o cero en caso contrario. Devuelve una columna nueva por cada columna de entrada, cada una de las cuales contiene la columna correspondiente en data1 comparada con la correspondiente columna de data2 (es decir, la primera columna de data1 se compara con la primera columna de data, la segunda columna con la segunda columna, etc.).

Si data2 es una constante, cada valor en data se compara con ese valor. Si data2 es una columna, los cálculos se realizan fila por fila. Los valores en data1 se comparan con el valor de la primera fila de data2, la segunda fila con la segunda fila y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** El operador GT se puede abreviar con un signo mayor que (>).

Este macro está disponible en IBM Interact.

## Ejemplos

TEMP = 3 GT 4 o TEMP = 3 > 4
Crea una nueva columna denominada TEMP que contiene el valor cero (ya que tres no es mayor que cuatro).
TEMP = V1 > 8
Crea una nueva columna denominada TEMP, donde cada valor es uno si el valor de fila correspondiente de la columna V1 es mayor que el número ocho, de lo contrario es cero.
TEMP = V1:V3 > 2
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son el contenido de la columna V1 comparados con el valor dos, los valores de la columna VX son el contenido de la columna V2 comparados con el valor dos y los valores de la columna VY son el contenido de la columna V3 comparados con el valor dos.
TEMP = V1 > V1
Crea una nueva columna denominada TEMP que contiene todos los ceros (ya que ningún número es mayor que sí mismo).
TEMP = V1 > V2
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 comparado con el valor de fila correspondiente de la columna V2.
TEMP = V1:V3 > V4:V6
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores en V1 comparados con los valores de fila correspondientes de la columna V4. La columna VX compara las columnas V2 y V5. La columna VY compara las columnas V3 y V6.

TEMP = V1[10:20] > V2 o TEMP = V1[10:20] > V2[1:11]

Crea una nueva columna denominada TEMP, donde las primeras once celdas contienen el resultado de comparar los valores de las filas 10 a 20 de la columna V1 con los valores de las filas 1 a 11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
EQ	Devuelve TRUE si un rango de datos es igual a otro
GE	Devuelve TRUE si un rango de datos es mayor o igual a otro
LE	Devuelve TRUE si un rango de datos es menor o igual a otro
LT	Devuelve TRUE si un rango de datos es menor que otro
NE	Devuelve TRUE si un rango de datos no es igual a otro

---

## HISTOGRAM



### Sintaxis

HISTOGRAM(data, bin\_col)

### Parámetros

data

El rango de celdas para calcular el histograma. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Todas las columnas en data deben ser del mismo tipo de datos (es decir, numérico o cadena de texto). Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

bin\_col

Los valores para los límites de las agrupaciones. Puede ser un valor constante, una columna, un rango de celdas de una sola columna o una expresión que evalúa a cualquiera de los anteriores. El tipo de datos de bin\_col debe ser igual al de data. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

HISTOGRAM calcula el histograma (es decir, la frecuencia de aparición de los valores de datos en diversas agrupaciones) de los valores del rango de datos especificado. Devuelve una única columna con el número de valores de datos de data que caen dentro del rango de agrupación correspondiente especificado por bin\_col.

Para valores numéricos, se forma una "agrupación" cada dos filas adyacentes de bin\_col. Cualquier valor de data que esté dentro de una agrupación es acumulado para esa agrupación. La columna de salida contiene el último recuento del número de valores de datos dentro de cada agrupación. El primer valor límite está *incluido* en la agrupación; el segundo está *excluido*. Por ejemplo, el par de valores límite 1 y 2 contendrán un recuento de todos los valores en data mayores o iguales a 1 y menores que 2. La longitud de la columna de salida es uno menos que la longitud de bin\_col.

Para cadenas de texto, solo cuentan en la agrupación las coincidencias exactas de la cadena de texto en bin\_col. La longitud de la columna de salida es la longitud de bin\_col. Para datos numéricos, si bin\_col es escalar (es decir, contiene un único valor de celda), se cuenta el número de elementos en data.

**Nota:** La función de macro HISTOGRAM coloca puntos de datos en agrupaciones en forma diferente que el gráfico de histograma de **IBM PredictiveInsight**. El gráfico del histograma excluye el mínimo (excepto por la agrupación más a la izquierda) e incluye el máximo de cada límite de agrupación.

## Ejemplos

TEMP = HISTOGRAM(1...10, COLUMN(1, 3, 10))
Crea una nueva columna denominada TEMP que contiene los valores 2 y 7.
TEMP = HISTOGRAM("a", "b", "a"), COLUMN("a", "b", "c"))
Crea una nueva columna denominada TEMP que contiene los valores 2, 1 y 0.
TEMP = HISTOGRAM(V1, COLUMN(1, 25, 50, 75, 101))
Crea una nueva columna denominada TEMP que contiene cuatro valores. El primer valor es el número de valores de la columna V1 mayores o iguales a 1 y menores que 25. El segundo valor es el número de valores en la columna V1 mayores o iguales a 25 y menores que 50. El tercer y el cuarto valor contienen los recuentos del tercer y cuarto cuartil respectivamente.
TEMP = HISTOGRAM(V1:V3, V4)
Crea una nueva columna denominada TEMP, donde cada valor es un recuento del número de valores en las columnas V1 a V3 que se encuentran dentro de los límites de la agrupación especificados por la columna V4.
TEMP = HISTOGRAM(V1[50:100]:V5, V6[1:10])
Crea una nueva columna denominada TEMP que contiene 10 valores. Cada valor es el número de valor de datos en las filas 50 a 100 de las columnas V1 a V5 que se encuentran dentro de los límites de la agrupación especificados por las filas 1 a 10 de la columna V6.

## Funciones relacionadas

Función	Descripción
COUNT	Cuenta la cantidad de celdas que contienen valores en el rango de datos especificado

---

# IF

## Sintaxis

IF(predicate\_col, then\_value) IF(predicate\_col, then\_value, else\_value)

## Parámetros

predicate\_col

Una columna de valores booleanos o una expresión que se evalúa en una sola columna de valores booleanos. Los valores booleanos se interpretan como cero o distinto de cero. Esta columna debe contener al menos tantas filas como el rango de datos del que se extraen los datos. De lo contrario, predicate\_col será un límite para el número de filas procesadas por la función de macro EXTRACT (consulte "Descripción" más abajo).

then\_value

El o los valores a devolver si la fila correspondiente de predicate\_col contiene un valor distinto de cero. Este puede ser un valor constante, una columna o una expresión que evalúa a cualquiera de los anteriores. Consulte "Parámetros de función de macro" en la página 9 para obtener la definición de formato de then\_value (al igual que data).

else\_value

Si se proporciona este parámetro opcional, es devuelto si la fila correspondiente de predicate\_col contiene un cero. Este puede ser un valor constante, una columna o una expresión que evalúa a cualquiera de los anteriores. Si no se proporciona else\_value, se devuelve un cero cuando predicate\_col evalúe en falso. Consulte "Parámetros de función de macro" en la página 9 para obtener la definición de formato de else\_value (al igual que data).

## Descripción

IF evalúa la expresión en predicate\_col y devuelve then\_value cuando la expresión es verdadera o else\_value si la expresión es falsa. Devuelve el mismo número de columnas en then\_value y else\_value. La o las columnas nuevas contienen el o los valores de fila correspondientes de then\_value si el valor de predicate\_col es distinto de cero. Si se proporciona else\_value, se devuelve cuando el valor de predicate\_col es cero. Si no se proporciona else\_value, se devuelve cero.

Debido a que IF opera fila por fila, produce un resultado para cada fila hasta el último valor de la columna más corta (es decir, la columna más corta de predicate\_col, then\_value y else\_value).

**Nota:** Por lo general, se usará para crear una columna de predicado utilizando una de las funciones de macro de comparación (por ejemplo, ==, >, <, ISEVEN, ISODD, etc.).

Este macro está disponible en IBM Interact.

## Ejemplos

<code>TEMP = IF(1, V1)</code> Crea una nueva columna denominada TEMP que contiene una copia de la columna V1.
<code>TEMP = IF(V1, 1, 0)</code> Crea una nueva columna denominada TEMP, donde cada valor es uno si el valor de fila correspondiente de la columna V1 es distinto de cero, de lo contrario es cero.
<code>TEMP = IF(V3, V1, V2)</code> Crea una nueva columna denominada TEMP, donde se copia cada valor de la columna V1 si el valor correspondiente de la columna V3 es distinto de cero; de lo contrario se copia el valor de la columna V2.
<code>TEMP = IF(ABS(V1-AVG(V1)) &lt; STDV(V1), V1)</code> Crea una nueva columna denominada TEMP que contiene cada valor de la columna V1 que esté menos de una desviación estándar separado del promedio.
<code>TEMP = IF(V3[20:30], V1[30:40], V2)</code> Crea una nueva columna denominada TEMP que contiene los valores para las filas 10 a 20. Cada valor es copiado de la columna V1 (celdas 10 a 20) si el valor correspondiente de la columna V3 (celdas 30 a 40) es distinto de cero; de lo contrario se copia el valor de la columna V2 (celdas 1 a 11).

## Funciones relacionadas

Función	Descripción
EXTRACT	Extrae filas de acuerdo a los valores de una columna de predicado
SELECT	Devuelve la(s) columna(s) especificada(s) de un rango de datos

---

## IN

### Sintaxis

`valuet IN (value1 AND value2 . . . .) or valuet IN subquery`

### Parámetros

El primer formato permite utilizar una lista de valores en lugar de una subconsulta.

El segundo formato utiliza una subconsulta que es evaluada para producir un resultado intermedio, contra el que se pueden realizar más procesos.

### Descripción

El predicado IN le permite usar una lista de valores en lugar de una subconsulta o introducirá una subconsulta.

**Nota:** IN se diferencia de ISMEMBER porque IN se lleva a cabo en la base de datos (si es posible), mientras ISMEMBER se calcula en el servidor.

**Nota:** El predicado IN tiene una versión negativa, NOT IN. El formato de esta es igual al de IN. NOT IN es verdadero únicamente si el valor proporcionado no se encuentra en los valores devueltos por la subconsulta.

Este macro está disponible en IBM Interact.

**Importante:** Cuando se utiliza IN en IBM Interact, puede utilizar sólo la sintaxis `value IN (value1 AND value2 . . . )` .

## Ejemplos

<pre>TEMP = IN(25, COLUMN(1...10))</pre>
Devuelve la(s) columna(s) especificada(s) de un rango de datos
<pre>TEMP = IN("cat", COLUMN("cat", "dog", "bird"))</pre>
Crea una nueva columna denominada TEMP que contiene el valor uno.
<pre>TEMP = IN(V1, V1)</pre>
Crea una nueva columna denominada TEMP que contiene todos unos.
<pre>TEMP = IN(V1, V2)</pre>
Crea una nueva columna denominada TEMP, donde cada valor es uno si la fila correspondiente de la columna V1 contiene un valor de la columna V2. De lo contrario, el valor es cero.

## Ejemplos

<pre>SELECT title, current_rental_price FROM movie_titles WHERE our_cost IN (14.95, 24.95, 29.95 ) ;</pre>
significa que todas las películas que cuestan \$14.95, \$24.95, o \$29.95 se evaluarán como VERDADERO para el predicado IN y, a su vez, produce una lista de títulos y adjunta los precios de alquiler actuales para esas películas.
<pre>UPDATE movie_titles SET current_rental_price = (regular_rental_price * .9) WHERE title IN ( SELECT movie_title FROM movie_stars WHERE actor_last_name = 'Stewart' AND actor_first_name = 'James' ) ;</pre>
Después de que la subconsulta produce la lista de títulos de películas, se procesa la lista contra la tabla MOVIE_TITLES y se descuentan los precios del alquiler adecuados para <i>Philadelphia Story</i> , <i>It's a Wonderful Life</i> , etc.

---

## INIT



### Sintaxis

```
INIT(val1 [, val2]...) INIT(column)
```

### Parámetros

val1

El valor numérico de la función recursiva en el tiempo (*t*-1).

val2

El valor numérico de la función recursiva en el tiempo ( $t-n$ ), donde  $n$  es el número de parámetro. Se puede repetir este parámetro varias veces para proporcionar valores iniciales para un número indefinido de pasos de tiempo anteriores.

column

Una columna de valores numéricos. Se asignará la primera celda al paso de tiempo ( $t-1$ ), la segunda celda a ( $t-2$ ), etc.

## Descripción

INIT especifica los valores iniciales para una definición de función recursiva. Se asigna el primer valor proporcionado al paso de tiempo  $t-1$ , el segundo a  $t-2$ , etc. Si no se inicializa un paso de tiempo utilizando INIT, se asume que su valor es cero. Por ejemplo, dada la sentencia

V1 = INIT(1, 2, 3)

el valor para el paso de tiempo ( $t-4$ ) es cero (como lo son todos los otros pasos de tiempo cada vez más atrás en el tiempo). Se necesita una sentencia INIT antes de definir cualquier función recursiva.

**Nota:** Para inicializar todos los valores a cero, puede especificar simplemente INIT() sin argumentos.

La función de macro INIT no devuelve ningún valor. Si se utiliza sola en una definición de función, devuelve una columna en blanco.

## Ejemplos

TEMP = INIT() t = 1 to 10 TEMP = 1 + TEMP[t-1]
Crea una nueva columna denominada TEMP que contiene los valores 1 a 10.
TEMP = INIT(1) t = 1 TO 100 TEMP = TEMP[t-1]+TEMP[t-1]
Crea una nueva columna denominada TEMP que contiene los valores 2, 4, 8, 16, 32, etc. Las primeras 100 celdas de TEMP contendrán valores.
TEMP = INIT(1, 2, 3, 4, 5) t = 1 to 500 TEMP = TEMP[t-5]
Crea una nueva columna denominada TEMP que contiene la secuencia de valores 5, 4, 3, 2, 1 repetida 100 veces.
TEMP = INIT(1, 2, 3) t = 1 to 1000 TEMP = 2*TEMP[t-1] + 4*TEMP[t-2]^2 - TEMP[t-3]
Crea una nueva columna denominada TEMP que contiene valores para la función recursiva:
$TEMP = 2*TEMP(t-1) + 4*TEMP(t-2)^2 - TEMP(t-2)$
Se calculan 1000 valores de celdas.

## Funciones relacionadas

Función	Descripción
T0	Genera operadores de rango



# INT

## Sintaxis

INT(data)

## Parámetros

data

Los valores numéricos a redondear a un valor entero. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

INT calcula el mayor entero menor que los valores (también conocido como el suelo) en el rango de datos especificado. INT devuelve una columna nueva por cada columna de entrada, cada una de las cuales contiene el suelo de números de la correspondiente columna de entrada.

**Nota:** Es lo mismo que la función de macro FLOOR.

## Ejemplos

TEMP = INT(4.7)
Crea una nueva columna denominada TEMP que contiene el valor 4.
TEMP = INT(-1.5)
Crea una nueva columna denominada TEMP que contiene el valor -2.
TEMP = INT(V1)
Crea una nueva columna denominada TEMP, donde cada valor es el mayor entero menor o igual a los contenidos en la columna V1.
TEMP = V1 - INT(V1)
Crea una columna nueva TEMP que contiene la décima proporción de cada valor en la columna V1.
TEMP = INT(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son los mayores enteros menores e iguales a los contenidos en la columna V1, los valores de la columna VX son los mayores enteros menores o iguales a los contenidos en la columna V2 y los valores de la columna VY son los mayores enteros menores o iguales a los contenidos en la columna V3.
TEMP = INT(V1[10:20])
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los mayores enteros menores o iguales al valor correspondiente en las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.

```
TEMP = INT(V1[1:5]:V2)
```

Crea dos nuevas columnas denominadas TEMP y VX, con valores en las filas 1-5 (las otras celdas están vacías). Los valores de la columna TEMP son los mayores enteros menores o iguales a los valores correspondientes de la columna V1 y los valores de la columna VX son los mayores enteros menores o iguales a los correspondientes valores de fila de la columna V2.

## Funciones relacionadas

Función	Descripción
ROUND	Calcula el valor redondeado del contenido del rango de datos especificado
TRUNCATE	Devuelve la parte no fraccionaria de cada valor del rango de datos especificado

---

## INTEGRAL



### Sintaxis

```
INTEGRAL(data [, Multiplicador])
```

### Parámetros

data

Los valores numéricos para calcular el integral de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

multiplier

Un valor por el cual multiplicar cada valor en data. Esta puede ser un valor constante o una expresión que evalúe a una constante.

### Descripción

INTEGRAL calcula el integral de los valores en un momento de la serie. Cada valor es la suma de todos los valores anteriores en el tiempo. Si se proporciona un valor para multiplier, cada valor se multiplica por el valor especificado. INTEGRAL devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene una integral de los valores de la correspondiente columna de entrada.

### Ejemplos

```
TEMP = INTEGRAL(5)
```

Crea una nueva columna denominada TEMP que contiene el valor 5.

TEMP = INTEGRAL(COLUMN(1,2,3))
Crea una nueva columna denominada TEMP que contiene los valores 1, 3y 6.
TEMP = INTEGRAL(COLUMN(1,2,3), 2)
Crea una nueva columna denominada TEMP que contiene los valores 2, 6y 12.
TEMP = INTEGRAL(V1)
Crea una nueva columna denominada TEMP, donde cada valor es la suma de todas las celdas anteriores en la columna V1.
TEMP = INTEGRAL(V1, 10)
Crea una nueva columna denominada TEMP, donde cada valor es la suma de todas las celdas anteriores en la columna V1 multiplicada por 10.
TEMP = INTEGRAL(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son las integrales de la columna V1, los valores de la columna VX son las integrales de la columna V2 y los valores de la columna VY son las integrales de la columna V3.
TEMP = INTEGRAL(V1:V3)
Crea una nueva columna denominada TEMP, donde las primeros 11 celdas contienen las integrales de los correspondientes valores en las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = INTEGRAL(V1:V3)
Crea dos nuevas columnas denominadas TEMP y VX cada una con valores en filas 1-5 (las otras celdas están vacías). Los valores de la columna TEMP son las integrales de los correspondientes valores de fila de la columna V1 y los valores de la columna VX son las integrales de los correspondientes valores de fila de la columna V2.

## Funciones relacionadas

Función	Descripción
DTEMP = INTEGRAL(V1:V3)	Calcula la derivada de los valores del rango de datos especificado
SUM o TOTAL	Calcula la suma de un rango de celdas

---

## INVERSE

### Sintaxis

INVERSE(data)

### Parámetros

data

Los valores numéricos para calcular el inverso de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

INVERSE calcula el valor negativo de los valores en el rango de datos especificado. Devuelve  $-x$  (es decir, se devuelven valores negativos como valores positivos y se devuelven valores positivos como valores negativos). INVERSE devuelve una nueva columna por cada columna de entrada, cada una de las cuales contiene los valores inversos a los de la columna de entrada correspondiente.

**Nota:** Para invertir un valor o una columna, agréguele un signo menos (-) delante. Por ejemplo,  $V2 = -V1$  es lo mismo que  $V2 = \text{INVERSE}(V1)$ .

## Ejemplos

<code>TEMP = INVERSE(3.2)</code>
Crea una nueva columna denominada TEMP que contiene el valor -3.2.
<code>TEMP = INVERSE(V1)</code>
Crea una nueva columna denominada TEMP, donde cada valor es el valor negativo de los valores en la columna V1.
<code>TEMP = INVERSE(V1:V3)</code>
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son los negativos de la columna V1, los valores de la columna VX son los negativos de la columna V2 y los valores de la columna VY son los negativos de la columna V3.
<code>TEMP = INVERSE(V1[10:20])</code>
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los valores negativos de los correspondientes valores en las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.
<code>TEMP = INVERSE(V1[1:5]:V2)</code>
Crea dos columnas nuevas denominadas TEMP y VX, cada una con valores en las filas 1 a 5 (las otras celdas están vacías). Los valores de la columna TEMP son los negativos de los valores de las filas correspondientes de la columna V1 y los valores de la columna VX son los negativos de los valores de las filas correspondientes de la columna V2.

## Funciones relacionadas

Función	Descripción
ABS	Calcula el número NOT lógico del contenido del rango de datos especificado valor absoluto del contenido del rango de datos especificado
NOT	Calcula el número NOT lógico del contenido del rango de datos rango de datos especificado
SIGN	Calcula el signo (positivo o negativo) de los valores del rango de datos especificado

---

## IS

### Sintaxis

IS <keyword>

## Parámetros

keyword

Condición de búsqueda, normalmente "NULL," "TRUE," "UNKNOWN," y "FALSE."

## Descripción

IS se utiliza en condiciones de búsqueda complejas. Cuanto más compleja la búsqueda, más útil puede ser la condición IS. Estas condiciones de búsqueda booleana proporcionan una forma alternativa de expresar las condiciones de búsqueda básicas.

Esta macro está disponible en IBM Interact.

IS devuelve diferentes resultados en IBM Interact desde IBM Campaign. NULL devuelve 1 si hay al menos un valor NULL para una ID de audiencia. UNKNOWN devuelve 1 para una ID de audiencia si esta no tiene ningún valor.

## Ejemplos

```
SELECT customer FROM customer_table1 WHERE (last_name = "Smith" AND first_name = "John") IS TRUE ;
```

generará una lista de todos los clientes se llaman John Smith.

```
SELECT customer FROM customer_table1 WHERE (last_name = "X" AND first_name = "X") IS UNKNOWN ;
```

solicita cualquier valor no nulo.

```
SELECT cost FROM cost_table1 WHERE (current_cost = "200" IS FALSE ;
```

enumera todos los valores de la tabla de costos que no sean \$200.

---

## ISERROR

### Sintaxis

ISERROR(data)

### Parámetros

data

Los valores a probar si cualquiera de las filas contiene un error (es decir, una celda que contenga el símbolo ??? . Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúe a cualquiera de los anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

ISERROR comprueba si alguna celda de cada fila del rango de datos especificado contiene un error (es decir, una celda que contenga el símbolo ??? . Devuelve una nueva columna donde cada fila contiene un uno si la fila correspondiente de data contiene un error. De lo contrario, contiene un cero. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más larga.

**Nota:** Esta función es útil para detectar errores en una columna y luego utilizar la función de macro EXTRACT para extraer las filas de datos sin errores.

## Ejemplos

TEMP = ISERROR(-3)
Crea una nueva columna denominada TEMP que contiene el valor cero.
TEMP = ISERROR(V1)
Crea una nueva columna denominada TEMP, donde cada valor es uno si la fila correspondiente de la columna V1 contiene el símbolo ???. De lo contrario, el valor es cero.
TEMP = ISERROR(V1:V3)
Crea una nueva columna denominada TEMP, donde cada valor es uno si cualquiera de las celdas de las filas correspondientes de las columnas V1 a V3 contiene el símbolo???. De lo contrario, el valor es cero.
TEMP = ISERROR(V1[50:100]:V10)
Crea una nueva columna denominada TEMP, con valores en las filas 1 a 50. Cada valor es un uno si alguna de las celdas en las filas 50 a 100 de las columnas V1 a V10 contiene el símbolo ???. De lo contrario, el valor es cero.

## Funciones relacionadas

Función	Descripción
EXTRACT	Extrae filas de acuerdo a los valores de una columna de predicado

---

## ISEVEN

### Sintaxis

ISEVEN(data)

### Parámetros

data

Los valores numéricos para comprobar si son pares. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

ISEVEN evalúa cada valor en el conjunto de datos especificado para comprobar los valores pares. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene un uno para todos los valores pares (es decir, el valor del módulo dos es cero) o un cero para todos los valores impares.

**Nota:** Para valores no enteros, se aplica primero la función de macro INT. Por ejemplo, ISEVEN (2.5) = 1, ya que 2 es par.

## Ejemplos

TEMP = ISEVEN(-3)
Crea una nueva columna denominada TEMP que contiene el valor cero.
TEMP = ISEVEN(MERGE(3, 2, 0))
Crea tres nuevas columnas denominadas TEMP, VX y VY. TEMP contiene el valor 0, VX contiene el valor 1 y VY contiene el valor 1.
TEMP = ISEVEN(V1)
Crea una nueva columna denominada TEMP, donde cada valor es el resultado de la comprobación de los valores pares del contenido de la columna V1.
TEMP = ISEVEN(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son los resultados de la comprobación de los valores pares del contenido de la columna V1, los valores de la columna VX son los resultados de comprobación de los valores pares de la columna V2 y los valores de la columna VY son los resultados de la comprobación de los valores pares del contenido de la columna V3.
TEMP = ISEVEN(V1[10:20])
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los resultados de la comprobación de valores pares de los valores de las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = ISEVEN(V1[1:5]:V2)
Crea dos nuevas columnas denominadas TEMP y VX cada una con valores en filas 1-5 (las otras celdas están vacías). Los valores de la columna TEMP son los resultados de la comprobación de valores pares de las filas correspondientes de la columna V1 y los valores de la columna VX son los resultados de la comprobación de valores pares de las filas correspondientes de la columna V2.

## Funciones relacionadas

Función	Descripción
ISODD	Comprueba si los valores de entrada son impares (es decir, no divisibles por dos)
ISMEMBER	Comprueba un rango de entrada en una "tabla" de valores, devolviendo un uno si un valor está contenido en la tabla. De lo contrario, devuelve cero.

---

## ISMEMBER



### Sintaxis

ISMEMBER(data, table)

### Parámetros

data

Los valores para comprobar si pertenecen a la tabla. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Si data contiene más de una columna, todas las columnas deben contener el mismo tipo de datos (numéricos o cadenas de texto). Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

table

Los valores de la tabla con los que realizar la comparación. Esto puede ser un valor constante, una columna, un rango de celdas, o una expresión para evaluar cualquiera de las anteriores. Si table contiene más de una columna, todas las columnas deben contener el mismo tipo de datos que table (numéricos o cadenas de texto). El número de valores de datos en table no puede superar los 16 millones. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

**Nota:** ISMEMBER difiere de IN porque ISMEMBER se calcula en el servidor, mientras que IN se realiza en la base de datos siempre que sea posible.

## Descripción

ISMEMBER compara los valores de datos en el rango de datos especificado con una tabla de valores de datos. Devuelve una nueva columna para cada columna de entrada en data, cada una de las cuales contiene un uno si el valor de entrada correspondiente pertenece a table, de lo contrario devuelve un cero.

## Ejemplos

<pre>TEMP = ISMEMBER(25, COLUMN(1...10))</pre> <p>Crea una nueva columna denominada TEMP que contiene el valor cero.</p>
<pre>TEMP = ISMEMBER("cat", COLUMN("cat", "dog", "bird"))</pre> <p>Crea una nueva columna denominada TEMP que contiene el valor uno.</p>
<pre>TEMP = ISMEMBER(V1, V1)</pre> <p>Crea una nueva columna denominada TEMP que contiene solo unos.</p>
<pre>TEMP = ISMEMBER(V1, V2)</pre> <p>Crea una nueva columna denominada TEMP, donde cada valor es uno si la fila correspondiente de la columna V1 contiene un valor de la columna V2. De lo contrario, el valor es cero.</p>
<pre>TEMP = ISMEMBER(V1:V2, V5:V10)</pre> <p>Crea dos nuevas columnas denominadas TEMP y VX. La columna TEMP contiene un uno si la fila correspondiente de la columna V1 pertenece a las columnas V5 a V10. De lo contrario, devuelve un cero. La columna VX P contiene un uno si la fila correspondiente de la columna V2 pertenece a las columnas V5 a V10. De lo contrario, devuelve un cero.</p>
<pre>TEMP = ISMEMBER(V1[10:15]:V2, V3[1:100]:V6)</pre> <p>Crea dos columnas denominadas TEMP y VX, cada una con valores en las filas 1 a 6 (las otras celdas están vacías). Los valores de la columna TEMP son uno si el contenido de las filas 10-15 de la columna V1 pertenecen a las filas 1 a 100 de las columnas V3 a V6. Los valores de la columna VX son uno si el contenido de las filas 10 a 15 de la columna V2 pertenecen a las filas 1 a 100 de las columnas V3 a V6.</p>



## Funciones relacionadas

Función	Descripción
ISEVEN	Comprueba si los valores de entrada son pares (es decir, divisibles por dos)
ISODD	Comprueba si los valores de entrada son impares (es decir, no divisibles por dos)

---

## ISODD

### Sintaxis

ISODD(data)

### Parámetros

data

Los valores numéricos para comprobar si son impares. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

ISODD evalúa cada valor en el conjunto de datos especificado para comprobar los valores impares. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene un uno para todos los valores impares (es decir, el valor del módulo dos es cero) o un cero para todos los valores no impares (es decir, los valores pares).

**Nota:** Para valores no enteros, se aplica primero la función de macro INT. Por ejemplo, ISODD(2.5) = 0, ya que 2 no es impar.

### Ejemplos

TEMP = ISODD(-3)
Crea una nueva columna denominada TEMP que contiene el valor uno.
TEMP = ISODD(MERGE(1, 4, 0))
Crea tres nuevas columnas denominadas TEMP, VX y VY. TEMP contiene el valor 1, VX contiene el valor 0 y VY contiene el valor 0.
TEMP = ISODD(V1)
Crea una nueva columna denominada TEMP, donde cada valor es el resultado de la comprobación de los valores impares del contenido de la columna V1.

TEMP = ISODD(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son los resultados de la comprobación de los valores impares del contenido de la columna V1, los valores de la columna VX son los resultados de comprobación de los valores impares de la columna V2 y los valores de la columna VY son los resultados de la comprobación de los valores impares del contenido de la columna V3.
TEMP = ISODD(V1[10:20])
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los resultados de la comprobación de valores impares de los valores de las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = ISODD(V1[1:5]:V2)
Crea dos columnas nuevas denominadas TEMP y VX, cada una con valores en las filas 1 a 5 (las otras celdas están vacías). Los valores de la columna TEMP son los resultados de la comprobación de valores impares de las filas correspondientes de la columna V1 y los valores de la columna VX son los resultados de la comprobación de valores impares de las filas correspondientes de la columna V2.

## Funciones relacionadas

Función	Descripción
ISEVEN	Comprueba si los valores de entrada son pares (es decir, divisibles por dos)
ISMEMBER	Comprueba un rango de entrada en una "tabla" de valores, devolviendo un uno si un valor está contenido en la tabla. De lo contrario, devuelve cero.

---

## CURTOSIS



### Sintaxis

KURTOSIS(data [, keyword])

### Parámetros

data

Los valores numéricos de los se calcula la curtosis. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de la función de macro" en el capítulo de esta guía para su producto IBM .Debe haber como mínimo cuatro valores en data.

keyword

Esta palabra clave opcional determina como se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las siguientes opciones:

ALL

Realiza el cálculo en todas las celdas en data (valor predeterminado)

COL

Realiza el cálculo por separado para cada columna de data

ROW

Realiza el cálculo por separado para cada fila de data

Para más detalles sobre el uso de palabras clave en IBM Campaign, vea “Especificaciones de formato” en la página 9.

Para más detalles sobre el uso de palabras clave en IBM PredictiveInsight, vea “Especificaciones de formato” en la página 27.

## Descripción

KURTOSIS calcula la curtosis de los valores en un rango de datos especificado. La curtosis es una medición del grado de apuntamiento o aplastamiento relativo de una determinada distribución en comparación con una distribución normal. Cuanto más negativa la curtosis, más plana es la distribución. Cuanto más positiva la curtosis, más apuntado es el pico de la distribución.

La curtosis se calcula de la manera siguiente:

$$\left\{ \frac{n(n+1)}{(n-1)(n-2)(n-3)} \sum_{j=1}^n \left( \frac{x_j - \text{mean}}{\sigma} \right)^4 \right\} - \frac{3(n-1)^2}{(n-2)(n-3)}$$

donde  $n$  es el número de muestras en la distribución,  $\text{mean}$  es el promedio y  $\sigma$  es la desviación estándar de la distribución. Debe proporcionarse un mínimo de tres valores de datos para calcular el coeficiente de curtosis.

## Ejemplos

TEMP = KURTOSIS(MERGE(3, 4, 5)) o TEMP = KURTOSIS(MERGE(3, 4, 5), ALL)
Crea una nueva columna denominada TEMP que contiene el valor -1.5.
TEMP = KURTOSIS(V1)
Crea una nueva columna denominada TEMP que contiene un valor único que es la curtosis del contenido de la columna V1.
TEMP = KURTOSIS(V1:V3)
Crea una nueva columna denominada TEMP que contiene un valor único que es la curtosis del contenido de las columnas V1, V2 y V3.
TEMP = KURTOSIS(V1[10:20])
Crea una nueva columna denominada TEMP que contiene un valor único que es la curtosis de las celdas de las filas 10 a 20 de la columna V1.

TEMP = KURTOSIS(V1[1:5]:V4)
Crea una nueva columna denominada TEMP que contiene un valor único que es la curtosis de las celdas de las filas 1 a 5 de las columnas V1 a V4.
TEMP = KURTOSIS(V1:V3, COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY. El valor único en la columna TEMP es la curtosis del contenido de la columna V1, el valor único en la columna VX es la curtosis del contenido de la columna V2 y el valor único en la columna VY es la curtosis del contenido de la columna V3.
TEMP = KURTOSIS(MERGE(1,4), COL)
Crea dos nuevas columnas denominadas TEMP y VX, cada una de las cuales contiene el valor -3.
TEMP = KURTOSIS(V1[1:5]:V3, COL) o TEMP = KURTOSIS(V1[1:5]:V3[1:5], COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene un valor único. El valor único en la columna TEMP es la curtosis de las celdas en las filas 1 a 5 de la columna V1, el valor único en la columna VX es la curtosis de las celdas en las filas 1 a 5 de la columna V2 y el valor único en la columna VY es la curtosis de las celdas en las filas 1 a 5 de la columna V3.
TEMP = KURTOSIS(V1:V3, ROW)
Crea una nueva columna denominada TEMP donde el valor en cada celda es la curtosis de la fila correspondiente en las columnas V1, V2 y V3.
TEMP = KURTOSIS(V1[1:5]:V3, ROW) o TEMP = KURTOSIS(V1[1:5]:V3[1:5], ROW)
Crea una nueva columna denominada TEMP donde las celdas en las filas 1 a 5 contienen la curtosis de la fila correspondiente en las columnas V1 a V3. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
STAT	Calcula desde el primer al cuarto momento del rango de datos especificado
SKEW	Calcula el desvío de la distribución de un rango de celdas
STDV o STDEV	Calcula la desviación estándar de un rango de celdas
VARIANCE	Calcula la varianza de un rango de celdas

## LAG



### Sintaxis

LAG(lag, data)

### Parámetros

lag

El número de incrementos temporales para el retardo. Este valor debe ser un entero positivo.

data

Los valores de retardo. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

LAG devuelve valores en el rango de datos de entrada con un retardo correspondiente al número especificado de incrementos temporales. Considera cada columna de entrada como una serie de datos en el tiempo y devuelve una nueva columna para cada columna de entrada. Cada nueva columna contiene los valores retardados por incrementos temporales (retardados el número lag de incrementos temporales) de los números en la columna de entrada correspondiente. Los primeros valores lag de la() columna(s) devuelta(s) son cero. La longitud de la(s) columna(s) devuelta(s) es la(s) longitud(es) de la(s) correspondiente(s) columna(s) de entrada + lag.

**Nota:** La función de macro LAG devuelve una columna con valores de tal manera que la celda VY[x ] = data[x-lag] para x > lag, else 0.

## Ejemplos

<code>TEMP = LAG(1, COLUMN(1,2,3,4))</code>
Crea una nueva columna denominada TEMP que contiene los valores 0,1, 2, 3 y 4 en las celdas 1 a 5 respectivamente.
<code>TEMP = LAG(2, V1)</code>
Crea una nueva columna denominada TEMP, donde cada valor es el contenido de la columna V1 retardado dos incrementos temporales.
<code>TEMP = LAG(10, V1:V3)</code>
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son el contenido de la columna V1 retardados diez incrementos temporales, los valores de la columna VX son el contenido de la columna V2 retardados diez incrementos temporales y los valores de la columna VY son el contenido de la columna V3 retardados diez incrementos temporales.
<code>TEMP = LAG(5, V1[10:20])</code>
Crea una nueva columna denominada TEMP, donde las primeras 5 celdas contienen 0, seguidas de las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.
<code>TEMP = LAG(2, V1[1:5]:V2)</code>
Crea dos columnas denominadas TEMP y VX, cada una con valores en las filas 1 a 7 (las otras celdas están vacías). Los valores en las filas 1 y 2 de cada columna son ceros. Los restantes valores en TEMP son los valores en las filas 1 a 5 de la columna V1. Los restantes valores en VX son las filas 1 a 5 de la columna V2.

## Funciones relacionadas

Función	Descripción
DELAY	Devuelve los valores de las columnas de entrada, retardados un número especificado de incrementos de tiempo
SLIDE_WINDOW	Crea un patrón desde una ventana especificada y lo desliza hacia abajo para crear el siguiente patrón

---

## LE

### Sintaxis

```
data1 LE data2 data1 <= data2
```

### Parámetros

data1

El rango de celdas numéricas a comparar. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúe a cualquiera de los anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

data2

El/los número(s) con los que comparar todos los valores de la columna especificada. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

LE compara los dos rangos de datos especificados, devolviendo un uno si los valores del primer conjunto de datos son menores o iguales a los valores del segundo conjunto de datos o un cero en caso contrario. Devuelve un nueva columna para cada columna de entrada, cada una de las cuales contiene la correspondiente columna en data1 comparada con la columna correspondiente de data2 (es decir que la primera columna de data1 se compara con la primera columna de data, la segunda columna con la segunda columna, y a sí sucesivamente).

Si data2 es una constante, cada valor en data se compara con ese valor. Si data2 es una columna, los cálculos se realizan fila por fila. Los valores en data1 se comparan con el valor de la primera fila de data2, la segunda fila con la segunda fila y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** El operador LE puede abreviarse con un signo de "menor que" seguido por un signo de igual (<=).

Este macro está disponible en IBM Interact.

## Ejemplos

<code>TEMP = 4 LE 4</code> o <code>TEMP = 4 &lt;= 4</code>
Crea una nueva columna denominada TEMP que contiene el valor uno (dado que cuatro es igual a sí mismo).
<code>TEMP = V1 &lt;= 8</code>
Crea una nueva columna denominada TEMP, donde cada valor es uno si el correspondiente valor de fila de la columna V1 es menor o igual al número ocho. De contrario, el valor es cero.
<code>TEMP = V1:V3 &lt;= 2</code>
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son el contenido de la columna V1 comparados con el valor dos, los valores de la columna VX son el contenido de la columna V2 comparados con el valor dos y los valores de la columna VY son el contenido de la columna V3 comparados con el valor dos.
<code>TEMP = V1 &lt;= V1</code>
Crea una nueva columna denominada TEMP que contiene solo unos (dado que todo número es igual a sí mismo).
<code>TEMP = V1 &lt;= V2</code>
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 comparado con el valor de fila correspondiente de la columna V2.
<code>TEMP = V1[10:20] &lt;= V2</code> o <code>TEMP = V1[10:20] &lt;= V2[1:11]</code>
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los resultados de la comparación de los valores de las filas 10 a 20 de la columna V1 con los valores en las filas 1 a 11 V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
EQ	Devuelve TRUE si un rango de datos es igual a otro
GE	Devuelve TRUE si un rango de datos es mayor o igual a otro
GT	Devuelve TRUE si un rango de datos es mayor que otro
LT	Devuelve TRUE si un rango de datos es menor que otro
NE	Devuelve TRUE si un rango de datos no es igual a otro

---

## LIKE

### Sintaxis

`data1 [NOT] LIKE data2`

### Parámetros

`data1`

El rango de celdas para comparar. Este puede ser una cadena de texto o una expresión que evalúe a una cadena de texto. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

data2

El patrón de texto con el que comparar todos los valores de la columna especificada. Este puede ser una cadena de texto o una expresión que evalúe a una cadena de texto. El número de columnas en data2 debe ser igual al número de columnas en data1, a menos que data2 sea una constante. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

Un subrayado (\_) en data2 representa un carácter comodín que coincidirá con cualquier carácter único en data1. Un signo de porcentaje (%) coincidirá con cero o más caracteres en data1.

## Descripción

LIKE compara los dos rangos de datos especificados, devolviendo un uno si las cadenas coinciden o un cero si no. Devuelve un nueva columna para cada columna de entrada, cada una de las cuales contiene la correspondiente columna en data1 comparada con la columna correspondiente de data2 (es decir que la primera columna de data1 se compara con la primera columna de data2, la segunda columna con la segunda columna, y a sí sucesivamente).

Si data2 es una constante de tipo serie, cada cadena en data1 se compara con esa cadena. Si data2 es una columna, los cálculos se realizan fila por fila. La cadena de la primera fila en data1 se comparan con la cadena de la primera fila de data2, la segunda fila con la segunda fila, y así sucesivamente. Este cálculo fila por fila da un resultado para cada fila hasta la última cadena de la columna más corta.

Cuando se comparan cadenas, no se distingue entre mayúsculas y minúsculas (es decir que "Yes", "YES", "yes" y "yeS" se consideran iguales).

**Nota:** La macro LIKE tiene una versión negativa: NOT LIKE. El formato es idéntico al de LIKE. NOT LIKE devuelve un uno si la cadena en data1 no coincide con la plantilla definida por data2.

Este macro está disponible en IBM Interact.

## Ejemplos

```
TEMP = "gold" LIKE "gold"
```

Crea una nueva columna denominada TEMP que contiene el valor uno (dado que las dos cadenas coinciden).

```
TEMP = "No" LIKE "NO"
```

Crea una nueva columna denominada TEMP que contiene el valor uno (las comparaciones de cadenas no distinguen entre mayúsculas y minúsculas).



TEMP = V1 LIKE "gold%"
Crea una nueva columna denominada TEMP, donde cada valor es uno si el correspondiente valor de fila de la columna V1 es igual a la cadena "gold" seguida de cualquier número de caracteres. De lo contrario, cada valor es cero.
TEMP = V1 LIKE "g_ld"
Crea una nueva columna denominada TEMP, donde cada valor es uno si el correspondiente valor de fila de la columna V1 es igual a la cadena "g" seguida de cualquier carácter seguido de "ld". De lo contrario, cada valor es cero.
TEMP = V1 LIKE V1
Crea una nueva columna denominada TEMP que contiene solo unos (dado que todo número es igual a sí mismo).
TEMP = V1 LIKE V2
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 comparado con el valor de fila correspondiente de la columna V2.
TEMP = V1:V3 LIKE V4:V6
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene las cadenas en V1 comparadas con las cadenas de fila correspondientes de la columna V4. La columna VX compara las columnas V2 y V5. La columna VY compara las columnas V3 y V6.
TEMP = V1[10:20] LIKE V2 o TEMP = V1[10:20] LIKE V2[1:11]
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los resultados de la comparación de las cadenas en las filas 10 a 20 de la columna V1 con las filas 1 a 11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
EQ	Devuelve TRUE si un rango de datos es igual a otro
GE	Devuelve TRUE si un rango de datos es mayor o igual a otro
GT	Devuelve TRUE si un rango de datos es mayor que otro
LE	Devuelve TRUE si un rango de datos es menor o igual a otro
LT	Devuelve TRUE si un rango de datos es menor que otro
NE	Devuelve TRUE si un rango de datos no es igual a otro

---

## LN o LOG

### Sintaxis

LN(data) o LOG(data)

### Parámetros

data

Los valores numéricos de los que calcular el logaritmo natural. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

LN o LOG calcula el logaritmo natural de cada valor del rango de datos especificado. Devuelve una nueva columna por cada columna de entrada, cada una de las cuales contiene el logaritmo natural de los números en la columna de entrada correspondiente. Los logaritmos naturales se basan en la constante  $e = 2,7182818$ . LN es la inversa de la función de macro EXP.

**Nota:** Todos los valores en el rango de datos especificado deben ser mayores a cero. De lo contrario, se devuelve una celda en blanco para cada entrada no válida.

## Ejemplos

TEMP = LN(3) o TEMP = LOG(3)
Crea una nueva columna denominada TEMP que contiene el valor 1.099.
TEMP = LN(V1)
Crea una nueva columna denominada TEMP, donde cada valor es el logaritmo natural del contenido de la columna V1.
TEMP = LN(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los logaritmos naturales del contenido de la columna V1, los valores en la columna VX son los logaritmos naturales del contenido de la columna V2 y los valores en la columna VY son los logaritmos naturales del contenido de la columna V3.
TEMP = LN(V1[10:20])
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los logaritmos naturales de los valores en las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = LN(V1[1:5]:V2)
Crea dos columnas nuevas denominadas TEMP y VX, cada una con valores en las filas 1 a 5 (las otras celdas están vacías). Los valores de la columna TEMP son los logaritmos naturales de las correspondientes filas de la columna V1 y los valores de la columna VX son los logaritmos naturales de las correspondientes filas de la columna V2.

## Funciones relacionadas

Función	Descripción
EXP	Calcula el número natural (e) elevado al contenido de cada celda del rango de datos especificado
LOG2	Calcula el logaritmo en base 2 del contenido del rango de datos especificado
LOG10	Calcula el logaritmo en base 10 del contenido del rango de datos especificado

Función	Descripción
POW	Calcula un valor base elevado a la(s) potencia(s) exponencial(es) especificada(s)

## LOG2

### Sintaxis

LOG2(data)

### Parámetros

data

Los valores numéricos de los que calcular el logaritmo en base 2. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

LOG2 calcula el logaritmo en base 2 de los valores en el rango de datos especificado. Devuelve una nueva columna por cada columna de entrada, cada una de las cuales contiene el logaritmo en base 2 de los números en la columna de entrada correspondiente.

**Nota:** Todos los valores en el rango de datos especificado deben ser mayores a cero. De lo contrario, se devuelve una celda en blanco para cada entrada no válida.

### Ejemplos

<p>TEMP = LOG2(8)</p> <p>Crea una nueva columna denominada TEMP que contiene el valor tres.</p>
<p>TEMP = LOG2(V1)</p> <p>Crea una nueva columna denominada TEMP, donde cada valor es el logaritmo en base 2 del contenido de la columna V1.</p>
<p>TEMP = LOG2(V1:V3)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los logaritmos base 2 del contenido de la columna V1, los valores en la columna VX son los logaritmos base 2 del contenido de la columna V2 y los valores en la columna VY son el logaritmo en base 2 del contenido de la columna V3.</p>
<p>TEMP = LOG2(V1[10:20])</p> <p>Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los logaritmos base 2 de los valores en las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.</p>

```
TEMP = LOG2(V1[1:5]:V2)
```

Crea dos columnas nuevas denominadas TEMP y VX, cada una con valores en las filas 1 a 5 (las otras celdas están vacías). Los valores de la columna TEMP son los logaritmos base 2 de las correspondientes filas de la columna V1 y los valores de la columna VX son los logaritmos base 2 de las correspondientes filas de la columna V2.

## Funciones relacionadas

Función	Descripción
LN o LOG	Calcula el logaritmo natural del contenido del rango de datos especificado
LOG10	Calcula el logaritmo en base 10 del contenido del rango de datos especificado
POW	Potencia exponencial

---

## LOG10

### Sintaxis

LOG10(data)

### Parámetros

data

Los valores numéricos de los que calcular el logaritmo en base 10. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

LOG10 calcula el logaritmo en base 10 de los valores en el rango de datos especificado. Devuelve una nueva columna por cada columna de entrada, cada una de las cuales contiene el logaritmo en base 10 de los números en la columna de entrada correspondiente.

**Nota:** Todos los valores en el rango de datos especificado deben ser mayores a cero. De lo contrario, se devuelve una celda en blanco para cada entrada no válida.

### Ejemplos

```
TEMP = LOG10(100)
```

Crea una nueva columna denominada TEMP que contiene el valor dos.

```
TEMP = LOG10(V1)
```

Crea una nueva columna denominada TEMP, donde cada valor es el logaritmo en base 10 del contenido de la columna V1.

TEMP = LOG10(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los logaritmos base 10 del contenido de la columna V1, los valores en la columna VX son los logaritmos base 10 del contenido de la columna V2 y los valores en la columna VY son el logaritmo en base 10 del contenido de la columna V3.
TEMP = LOG10(V1[10:20])
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los logaritmos base 10 de los valores en las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = LOG10(V1[1:5]:V2)
Crea dos nuevas columnas denominadas TEMP y VX cada una con valores en filas 1-5 (las otras celdas están vacías). Los valores de la columna TEMP son los logaritmos base 10 de las correspondientes filas de la columna V1 y los valores de la columna VX son los logaritmos base 10 de las correspondientes filas de la columna V2.

## Funciones relacionadas

Función	Descripción
LN o LOG	Calcula el logaritmo natural del contenido del rango de datos especificado
LOG2	Calcula el logaritmo en base 2 del contenido del rango de datos especificado
POW	Potencia exponencial

---

## LOWER

### Sintaxis

LOWER(data)

### Parámetros

data

Valor de la cadena que se convertirá a minúsculas.

### Descripción

LOWER convierte cada valor de cadena en el rango de datos especificado a minúsculas. Devuelve una nueva columna en la que cada celda contiene la cadena en minúsculas de la correspondiente celda de entrada.

Este macro está disponible en IBM Interact.

### Ejemplos

Temp = LOWER "GOLD"
Crea una nueva columna denominada Temp que contiene la cadena "gold".

```
TEMP = LOWER( "15 DE ENERO DE 1997")
```

Crea una nueva columna denominada TEMP, que contiene la cadena de texto ASCII "15 de enero de 1997".

```
TEMP = LOWER( "Presión")
```

Crea una nueva columna denominada TEMP, que contiene la cadena de texto ASCII "presión".

```
TEMP = LOWER(V1)
```

Crea una nueva columna denominada TEMP que contiene los caracteres en minúsculas de cada cadena en la columna V1.

---

## LT

### Sintaxis

```
data1 LT data2 data1 < data2
```

### Parámetros

data1

El rango de celdas numéricas a comparar. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúe a cualquiera de los anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

data2

El/los número(s) con los que comparar todos los valores de la columna especificada. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

LT compara los dos rangos de datos especificados, devolviendo un uno si los valores del primer conjunto de datos son menores que los valores del segundo conjunto de datos, o un cero en caso contrario. Devuelve un nueva columna para cada columna de entrada, cada una de las cuales contiene la correspondiente columna en data1 comparada con la columna correspondiente de data2 (es decir que la primera columna de data1 se compara con la primera columna de data, la segunda columna con la segunda columna, y a sí sucesivamente).

Si data2 es una constante, cada valor en data se compara con ese valor. Si data2 es una columna, los cálculos se realizan fila por fila. Los valores en data1 se comparan con el valor de la primera fila de data2, la segunda fila con la segunda fila y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** El operador LT puede abreviarse con un signo de "menor que" (<).

Este macro está disponible en IBM Interact.

## Ejemplos

TEMP = 3 LT 4 o TEMP = 3 < 4
Crea una nueva columna denominada TEMP que contiene el valor uno (dado que tres es menor que cuatro).
TEMP = V1 < 8
Crea una nueva columna denominada TEMP, donde cada valor es uno si el correspondiente valor de fila de la columna V1 es menor que el número ocho. De lo contrario, el valor es cero.
TEMP = V1:V3 < 2
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son el contenido de la columna V1 comparados con el valor dos, los valores de la columna VX son el contenido de la columna V2 comparados con el valor dos y los valores de la columna VY son el contenido de la columna V3 comparados con el valor dos.
TEMP = V1 < V1
Crea una nueva columna denominada TEMP que contiene solo ceros (dado que ningún número es menor que sí mismo).
TEMP = V1 < V2
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 comparado con el valor de fila correspondiente de la columna V2.
TEMP = V1[10:20] < V2 o TEMP = V1[10:20] < V2[1:11]
Crea una nueva columna denominada TEMP, donde las once primeras celdas contienen los resultados de comparar los valores en las filas 10 a 20 de la columna V1 con las filas 1 a 11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
EQ	Devuelve TRUE si un rango de datos es igual a otro
GE	Devuelve TRUE si un rango de datos es mayor o igual a otro
GT	Devuelve TRUE si un rango de datos es mayor que otro
LE	Devuelve TRUE si un rango de datos es menor o igual a otro

---

## LTRIM

### Sintaxis

LTRIM(data)

## Parámetros

data

La cadena de la que se eliminará el espacio al principio.

## Descripción

LTRIM elimina los caracteres de espacio al principio de cada valor de cadena en el rango de datos especificado, devolviendo la cadena convertida. Devuelve una nueva columna para cada columna de entrada.

Este macro está disponible en IBM Interact.

## Ejemplos

```
Temp = LTRIM " gold"
```

Crea una nueva cadena llamada Temp que contiene "gold".

---

## MAX

### Sintaxis

```
MAX(data [, keyword])
```

### Parámetros

data

Los valores numéricos de los que calcular el máximo. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las opciones siguientes:

ALL - Realiza el cálculo en todas las celdas de data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de data

Para más detalles sobre el uso de palabras clave en IBM Campaign, vea "Especificaciones de formato" en la página 9.

Para más detalles sobre el uso de palabras clave en IBM PredictiveInsight, vea "Especificaciones de formato" en la página 27.

**Nota:** Varias funciones de macro toman los parámetros de la palabra clave {ALL | COL | ROW}. Estas palabras clave no son aplicables en IBM Campaign porque los datos de entrada son siempre una sola columna o campo. La macro siempre se



comportará como si se especificara la palabra clave COL. Por lo tanto, no es necesario especificar estas palabras clave al utilizar IBM Campaign.

## Descripción

MAX calcula el máximo de los valores en el rango de datos especificado. Devuelve una única nueva columna que contiene el valor máximo.

Este macro está disponible en IBM Interact.

## Ejemplos

<p>TEMP = MAX(3) o TEMP = MAX(3, ALL)</p> <p>Crea una nueva columna denominada TEMP que contiene el valor tres.</p>
<p>TEMP = MAX(SELECT(COLUMN(1,3,5), V1:V5))</p> <p>Crea una nueva columna denominada TEMP que contiene un valor único que es el máximo de las columnas V1, V3 y V5.</p>
<p>TEMP = MAX(V1)</p> <p>Crea una nueva columna denominada TEMP que contiene un valor único que es el máximo valor de los contenidos de la columna V1.</p>
<p>TEMP = MAX(V1:V3)</p> <p>Crea una nueva columna denominada TEMP que contiene un valor único que es el máximo de las columnas V1, V2 y V3.</p>
<p>TEMP = MAX(V1[10:20])</p> <p>Crea una nueva columna denominada TEMP que contiene un valor único que es el máximo de las celdas de las filas 10 a 20 de la columna V1.</p>
<p>TEMP = MAX(V1[1:5]:V4)</p> <p>Crea una nueva columna denominada TEMP que contiene un valor único que es el máximo de las celdas de las filas 1 a 5 de las columnas V1 a V4.</p>
<p>TEMP = MAX(V1:V3, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. El valor único en la columna TEMP es el máximo de los contenidos de la columna V1, el valor único en la columna VX es el máximo de los contenidos de la columna V2 y el valor único en la columna VY es el máximo de los contenidos de la columna V3.</p>
<p>TEMP = MAX(V1[1:5]:V3, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene un valor único. El valor único en la columna TEMP es el máximo de las celdas en las filas 1 a 5 de la columna V1, el valor único en la columna VX es la curtosis de las celdas en las filas 1 a 5 de la columna V2 y el valor único en la columna VY es el máximo de las celdas en las filas 1 a 5 de la columna V3.</p>
<p>TEMP = MAX(V1:V3, ROW)</p> <p>Crea una nueva columna denominada TEMP donde el valor en cada celda es el máximo de la fila correspondiente en las columnas V1, V2 y V3.</p>
<p>TEMP = MAX(V1[10:20]:V3, ROW)</p> <p>Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen el máximo de los valores en las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.</p>

## Funciones relacionadas

Función	Descripción
DECIMATE	Realiza la decimación de una columna de números en varias columnas donde un "uno" indica que el valor de índice.
MAXINDEX	Devuelve el índice de columna del n <sup>-simo</sup> (primero, segundo, tercero, etc.) valor máximo para cada fila de la columna especificada
MIN	Calcula el número mínimo de un rango de celdas

---

## MAXINDEX



### Sintaxis

MAXINDEX(data [, n])

### Parámetros

data

El inicio de un rango de datos para el que calcular el índice del n<sup>simo</sup> valor máximo para cada fila. Este puede ser una columna o una expresión que evalúe a una constante. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

n

Número, mayor que cero. El valor predeterminado es uno y devuelve el índice del valor más alto para cada fila. (Dos devuelve el índice para el segundo valor más alto, tres el índice para el tercer valor más alto, etc.)

### Descripción

MAXINDEX encuentra el n<sup>simo</sup> valor máximo en cada fila de rango de datos especificado y devuelve una índice de columna para su ubicación. Devuelve una nueva columna que contiene un valor único que representa el índice del n<sup>simo</sup> valor máximo para cada fila. Un valor de uno indica la primera celda de la primera columna. Si hay más de un valor máximo, se devuelve la primera columna que contiene el n<sup>simo</sup> máximo.

**Nota:** Si tiene varias columnas, cada una de las cuales representa una clase distinta de salida de datos, puede utilizar MAXINDEX para seleccionar la "clase ganadora". Después puede hacer la capacitación con una sola columna de salida en lugar de varias columnas de salida. Por ejemplo, V4 = MAXINDEX(V1:V3), haría lo siguiente:

```
V1 V2 V3 V4 0 1 0 2 1 0 0 1 0 0 1 3
```

## Ejemplos

TEMP = MAXINDEX(MERGE(3,5,-2))

Crea una nueva columna denominada TEMP que contiene el valor dos (dado que el valor máximo es cinco, lo que ocurre en la segunda columna).

TEMP = MAXINDEX(V1)

Crea una nueva columna denominada TEMP que contiene un uno para cada fila de la columna V1.

TEMP=MAXINDEX (V6:V8,3)

Crea una nueva columna denominada TEMP con cada valor que representa el índice del valor mínimo (en este caso el tercero más alto de 3) de la correspondiente fila de las columnas V6, V7 y V8. Se devuelve un uno si el mínimo está en la columna V6, un dos si el mínimo está en la columna V7 y un tres si el mínimo está en la columna V8

TEMP = MAXINDEX(V6:V8)

Crea una nueva columna denominada TEMP donde cada valor representa el índice del máximo valor de la fila correspondiente de las columnas V6, V7 y V8. Se devuelve un uno si el máximo está en la columna V6 y un tres si el máximo está en la columna V8.

TEMP = MAXINDEX(V1[1:5]:V3]

Crea una nueva columna denominada TEMP donde las celdas en las filas 1 a 5 contienen un valor que representa el índice del valor máximo de la fila correspondiente de las columnas V1 a V3. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
DECIMATE	Realiza la decimación de una columna de números en varias columnas donde un uno indica el valor de índice.
MAX	Calcula el número máximo de un rango de celdas
MIN	Calcula el número mínimo de un rango de celdas

---

## MEAN

### Sintaxis

MEAN(data [, keyword])

### Parámetros

data

Los valores numéricos de los que calcular la media aritmética. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las opciones siguientes:

ALL - Realiza el cálculo en todas las celdas de data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de data

Consulte "DATE" en la página 77 para obtener más detalles sobre la utilización de palabras clave.

**Nota:** Muchas funciones de macro toman parámetros de palabra clave {ALL | COL | ROW}. Estas palabras clave no se aplican en **IBM Campaign** porque los datos de entrada son siempre una columna o un campo único. La macro siempre se comportará como si se especificara la palabra clave COL. Por lo tanto, no necesita especificar estas palabras claves cuando utilice **IBM Campaign** .

## Descripción

MEAN calcula la media aritmética o el promedio de las celdas del rango de datos especificado. La media aritmética se calcula sumando el contenido de todas las celdas dividido por el número de celdas. El número de columnas devuelto por MEAN depende de keyword.

- Si keyword es ALL, MEAN devuelve una nueva columna, que contiene un valor único (el promedio de todas las celdas en data).
- Si keyword es COL, MEAN devuelve una nueva columna por cada columna de entrada. Cada columna nueva contiene un valor (el promedio de todas las celdas de la correspondiente columna de entrada).
- Si keyword es ROW, MEAN devuelve una nueva columna, que contiene el promedio de todas las filas en data).

**Nota:** Las celdas en blanco se ignoran en la media.

**Nota:** MEAN es igual a la función de macro AVG.

Este macro está disponible en IBM Interact.

## Ejemplos

TEMP = MEAN(MERGE(3, 4, 5)) o TEMP = MEAN(MERGE(3, 4, 5), ALL)
Crea una nueva columna denominada TEMP que contiene el valor 4.
TEMP = MEAN(MERGE(-10, 6, 10))
Crea una nueva columna denominada TEMP que contiene el valor 2.
TEMP = MEAN(V1)
Crea una nueva columna denominada TEMP que contiene un valor único que es la media aritmética del contenido de la columna V1.
TEMP = MEAN(V1:V3)
Crea una nueva columna denominada TEMP que contiene un valor único que es la media aritmética del contenido de las columnas V1, V2 y V3.

TEMP = MEAN(V1[10:20])
Crea una nueva columna denominada TEMP que contiene un valor único que es la media aritmética de las celdas de las filas 10 a 20 de la columna V1.
TEMP = MEAN(V1[1:5]:V4)
Crea una nueva columna denominada TEMP que contiene un valor único que es la media aritmética de las celdas de las filas 1 a 5 de las columnas V1 a V4.
TEMP = MEAN(V1:V3, COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY. El valor único en la columna TEMP es la media aritmética del contenido de la columna V1, el valor único en la columna VX es la media aritmética del contenido de la columna V2 y el valor único en la columna VY es la media aritmética del contenido de la columna V3.
TEMP = MEAN(MERGE(1,4), COL)
Crea dos nuevas columnas denominadas TEMP y VX. TEMP contiene un valor único de uno, VX contiene un valor único de cuatro.
TEMP = MEAN(V1[10:20]:V3, COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene un valor único. El valor único en la columna TEMP es la media aritmética de las celdas en las filas 10 a 20 de la columna V1, el valor único en la columna VX es la media aritmética de las celdas en las filas 10 a 20 de la columna V2 y el valor único en la columna VY es la media aritmética de las celdas en las filas 10 a 20 de la columna V3.
TEMP = MEAN(V1:V3, ROW)
Crea una nueva columna denominada TEMP donde el valor en cada celda es la media aritmética de la fila correspondiente de las columnas V1, V2 y V3.
TEMP = MEAN(V1[1:5]:V3, ROW)
Crea una nueva columna denominada TEMP donde las celdas en las filas 1 a 5 contienen la media aritmética de la fila correspondiente de las columnas V1 a V3. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
AVG_DEV	Calcula la desviación promedio de un rango de celdas
SUM o TOTAL	Calcula la suma de un rango de celdas

## MERGE



### Sintaxis

MERGE(data [, data]...) {data [, data]...}

### Parámetros

data

El nombre de una columna a combinar con un rango de datos. Este puede ser un valor constante (numérico o texto ASCII entrecorillado), una columna, un rango de celdas o una expresión que evalúe a cualquiera de los anteriores. Este parámetro puede repetirse varias veces. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

MERGE concatena horizontalmente su entrada en un nuevo grupo. Devuelve el mismo número de columnas que se proporcione como entrada. Puede proporcionarse un número ilimitado de argumentos.

**Nota:** La función de macro MERGE puede especificarse utilizando llaves ( {} ). Sólo hay que insertar los argumentos entre las llaves, separados por comas (por ejemplo, TEMP = {1,2,3} es equivalente a TEMP = MERGE(1,2,3)).

## Ejemplos

TEMP = MERGE(3, 4, "cinco") o TEMP = {3, 4, "cinco"}
Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene un los valores 3, 4 y "cinco " respectivamente.
TEMP = V1:V3 o TEMP = MERGE(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VXy VY, donde TEMP es una copia de la columna V1, VX es una copia de la columna V2y VY es una copia de la columna V3.
TEMP = MERGE(V1, V3, V5:V7)
Crea cinco nuevas columnas denominadas TEMP, VW, VX, VYy VZ. TEMP es una copia de la columna V1, VW es una copia de la columna V3, VX a VZ son copias de las columnas V5 a V7.
TEMP = AVG(MERGE(V1,V3,V5), ROW)
Crea una nueva columna denominada TEMP donde cada fila es el promedio de la fila correspondiente en las columnas V1, V3 y V5. Un promedio sólo se calcula hasta llegar a la última fila de la columna de entrada más corta.
TEMP = MERGE(V1[10:50],V3, V5:V7[1:30])
Crea cinco nuevas columnas denominadas TEMP, VW, VX, VYy VZ. TEMP es una copia de los valores en las filas 10 a 50 de la columna V1, VW es una copia de los valores en la columna V3, VX a VZ son copias de los valores en las filas 1 a 30 de las columnas V5 a V7.
TEMP = AVG(MERGE(V1, V5:V6)) o TEMP = AVG({V1, V5:V6})
Crea una nueva columna denominada TEMP donde cada fila es el promedio de todas las celdas en las columnas V1, V5 y V6.

## Funciones relacionadas

Función	Descripción
COLUMN	Crea columnas nuevas, concatenando verticalmente los valores de entrada en cada columna
SELECT	Devuelve la(s) columna(s) especificada(s) de un rango de datos
TRANSPOSE	Traslada un rango de datos especificado

## Sintaxis

MIN(data [, keyword])

## Parámetros

data

Los valores numéricos de los que calcular el mínimo. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las opciones siguientes:

ALL - Realiza el cálculo en todas las celdas de data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de data

Consulte "DATE" en la página 77 para obtener más detalles sobre la utilización de palabras clave.

**Nota:** Muchas funciones de macro toman parámetros de palabra clave {ALL | COL | ROW}. Estas palabras clave no se aplican en **IBM Campaign** porque los datos de entrada son siempre una columna o un campo único. La macro siempre se comportará como si se especificara la palabra clave COL. Por lo tanto, no necesita especificar estas palabras claves cuando utilice **IBM Campaign** .

## Descripción

MIN calcula el mínimo de todas las celdas en el rango de datos especificado. Devuelve una única nueva columna que contiene el valor mínimo.

Este macro está disponible en IBM Interact.

## Ejemplos

TEMP = MIN(MERGE(1,10,-2)) Crea una nueva columna denominada TEMP que contiene el valor -2.
TEMP = MIN(V1) Crea una nueva columna denominada TEMP que contiene un valor único que es el valor mínimo de la columna V1.
TEMP = MIN(V1:V3) Crea una nueva columna denominada TEMP que contiene un valor único que es el mínimo de las columnas V1, V2 y V3.

TEMP = MIN(V1[10:20])
Crea una nueva columna denominada TEMP que contiene un valor único que es el mínimo de las celdas de las filas 10 a 20 de la columna V1.
TEMP = MIN(V1[1:5]:V4)
Crea una nueva columna denominada TEMP que contiene un valor único que es el mínimo de las celdas de las filas 1 a 5 de las columnas V1 a V4.
TEMP = MIN(V1:V3, COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY. El valor único en la columna TEMP es el mínimo de la columna V1, el valor único en la columna VX es el mínimo de la columna V2 y el valor único en la columna VY es el mínimo de la columna V3.
TEMP = MIN(V1[1:5]:V3, COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene un valor único. El valor en la columna TEMP es el mínimo de las celdas en las filas 1 a 5 de la columna V1, el valor único en la columna VX es la curtosis de las celdas en las filas 1 a 5 de la columna V2 y el valor en la columna VY es el mínimo de las celdas en las filas 1 a 5 de la columna V3.
TEMP = MIN(V1:V3, ROW)
Crea una nueva columna denominada TEMP donde el valor en cada celda es el mínimo de la fila correspondiente en las columnas V1, V2 y V3.
TEMP = MIN(V1[10:20]:V3, ROW)
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen el mínimo de los valores en las filas 1 a 5 de las columnas V1 a V3. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
DECIMATE	Realiza la decimación de una columna de números en varias columnas donde un uno indica el valor de índice.
MAX	Calcula el número máximo de un rango de celdas
MAX_TO_INDEX	Devuelve el índice de columna de valor máximo para cada fila de la columna especificada

---

## MINUS

### Sintaxis

data MINUS subtrahend data - subtrahend

### Parámetros

data

El rango de celdas que contienen números de los que restar. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a



uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

subtrahend

El/los número(s) de los que restar todos los valores de la columna especificada. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en subtrahend debe ser igual al número de columnas en data, a menos que subtrahend sea una constante. Para obtener la definición de formato de subtrahend (igual a data), consulte la sección "Parámetros de la función de macro" en el capítulo de esta guía para su producto IBM .

## Descripción

MINUS resta subtrahend del rango de datos data. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene la correspondiente columna en data menos la columna correspondiente de subtrahend (es decir que a la primera columna de data se le resta la primera columna de subtrahend, a la segunda columna se le resta la segunda columna, y a sí sucesivamente).

Si subtrahend es una constante, a cada valor en data se le resta ese valor. Si subtrahend contiene una o más columnas, los cálculos se realizan fila por fila entre una columna de data y una columna de subtrahend. A la primera fila en data se le resta el primer valor de fila de subtrahend, a la segunda fila se le resta la segunda fila, y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** El operador MINUS puede abreviarse con un signo menos o un guión (-).

Este macro está disponible en IBM Interact.

## Ejemplos

$TEMP = 7 \text{ MINUS } 4$ o $TEMP = 7 - 4$
Crea una nueva columna denominada TEMP que contiene el valor tres.
$TEMP = V1 - 8$
Crea una nueva columna denominada TEMP, donde cada valor es el contenido de la columna V1 menos ocho.
$TEMP = V1:V3 - 2$
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son el contenido de la columna V1 menos dos, los valores de la columna VX son el contenido de la columna V2 menos dos y los valores de la columna VY son el contenido de la columna V3 menos dos.
$TEMP = V1 - V1$
Crea una nueva columna denominada TEMP que contiene solo ceros (dado que cualquier número restado a sí mismo da cero).
$TEMP = V1 - V2$
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 menos el correspondiente valor de fila de la columna V2.

TEMP = V1:V3 -V4:V6

Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores en V1 menos los valores de fila correspondientes de la columna V4. La columna VX resta la columna V5 de la V2. La columna VY resta la columna V6 de la V3.

TEMP = V1[10:20] - V2 o TEMP = V1[10:20] - V2[1:11]

Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los valores de las filas 10 a 20 de la columna V1 menos los valores en las filas 1 a 11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
PLUS	Suma el contenido de dos rangos de datos
SUM o TOTAL	Calcula la suma de un rango de celdas

---

## MOD

### Sintaxis

data MOD divisor data % divisor

### Parámetros

data

Los valores numéricos de los que calcular el módulo. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

divisor

El entero de base no cero respecto del cual calcular el módulo. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en divisor debe ser igual al número de columnas en data, a menos que divisor sea una constante. Para obtener la definición de formato de divisor (al igual que data), consulte la sección "Parámetros de la función de macro" en el capítulo de esta guía para su producto IBM .

### Descripción

MOD calcula el resto de la división del rango de datos especificado por un valor especificado. Se calcula dividiendo cada valor por divisor y devolviendo el resto. Devuelve una nueva columna por cada columna de entrada, cada una de las cuales contiene los números en data módulo divisor. El resto tendrá el mismo signo (positivo o negativo) que data.

Si divisor es una constante, cada valor en la columna especificada se calcula módulo ese valor. Si divisor es una columna, los cálculos se realizan fila por fila. Los valores en data se calculan módulo el primer valor de fila de divisor, la

segunda fila con la segunda fila, y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** Si divisor es cero, un error de división por cero se devuelven.

**Nota:** El operador MOD puede abreviarse con un signo de porcentaje (%). Por ejemplo,  $TEMP = 5 \% 3$  es equivalente a  $TEMP = 5 \text{ MOD } 3$ .

Este macro está disponible en IBM Interact.

## Ejemplos

<p><math>TEMP = 10 \text{ MOD } 8</math> o <math>TEMP = 10 \% 8</math></p> <p>Crea una nueva columna denominada TEMP que contiene el valor 2.</p>
<p><math>TEMP = -10 \% 8</math></p> <p>Crea una nueva columna denominada TEMP que contiene el valor -2.</p>
<p><math>TEMP = V1 \% 8</math></p> <p>Crea una nueva columna denominada TEMP, donde cada valor es el contenido de la columna V1 módulo ocho.</p>
<p><math>TEMP = V1:V3 \% 2</math></p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los valores módulo dos del contenido de la columna V1, los valores en la columna VX son los valores módulo dos del contenido de la columna V2 y los valores en la columna VY son los valores módulo dos del contenido de la columna V3.</p>
<p><math>TEMP = V1 \% V1</math></p> <p>Crea una nueva columna denominada TEMP que contiene un cero para cada entrada de la columna V1. Esto se debe a que módulo de cada número en sí mismo es cero.</p>
<p><math>TEMP = V1 \% V2</math></p> <p>Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 módulo el correspondiente valor de fila de la columna V2. Tenga en cuenta que si <math>V2=V1</math>, entonces se devuelven solo ceros, como en el ejemplo anterior.</p>
<p><math>TEMP = V1:V3 \% V4:V6</math></p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores en V1 módulo los valores de fila correspondientes de la columna V4. La columna VX contiene los resultados de la columna V2 módulo V5. La columna VY contiene los resultados de la columna V3 módulo V6.</p>
<p><math>TEMP = V1[10:20] \% V2</math> o <math>TEMP = V1[10:20] \% V2[1:11]</math></p> <p>Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los valores de las filas 10 a 20 de la columna V1 módulo los valores en las filas 1 a 11 de la columna V2. Las otras celdas de TEMP están vacías.</p>

## Funciones relacionadas

Función	Descripción
DIV	Divide un rango de datos especificado por otro
MOD	Calcula el módulo del contenido del rango de datos especificado

---

## MONTHOF

### Sintaxis

MONTHOF(date\_string [, input\_format])

### Parámetros

date\_string

Un texto que representa una fecha válida.

input\_format

Una de las palabras claves en la tabla siguiente que especifica el formato de fecha de date\_string.

### Descripción

MONTHOF devuelve el mes como un número para la fecha especificada por la date\_string. Si no se proporciona input\_format, se utilizará la palabra clave predeterminada DELIM\_M\_D\_Y.

### Ejemplos

MONTHOF("012171",MMDDYY) devuelve el número 1.

**Nota:** Consulte "DATE" en la página 77 para obtener más información sobre formatos de fecha válidos.

### Funciones relacionadas

Función	Descripción
DAYOF	Devuelve el día de la semana como número.
WEEKDAYOF	Devuelve el día de la semana como número.
YEAROF	Devuelve el año como número.

---

## MULT

### Sintaxis

data MULT multiplier data \* multiplier

### Parámetros

data

Los valores numéricos a multiplicar. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúe a cualquiera de los anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

multiplier

El por el cual multiplicar todos los valores de la columna especificada. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en multiplier debe ser igual al número de columnas en data, a menos que multiplier sea una constante. Para obtener la definición de formato de multiplier (al igual que data), consulte la sección "Parámetros de la función de macro" en el capítulo de esta guía para su producto IBM .

## Descripción

MULT multiplica los valores de los dos rangos de datos especificados. Devuelve una nueva columna por cada columna de entrada, cada una de las cuales contiene los números en data multiplicados por multiplier. Si multiplier es una constante, cada valor en data se multiplica por este valor. Si multiplier es una columna, los cálculos se realizan fila por fila. Los valores en data se multiplican por el primer valor de fila de multiplier, la segunda fila con la segunda fila, y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** Utilizar una columna que contiene el mismo número  $x$  en cada fila como multiplier es lo mismo que utilizar la constante  $x$  como multiplier.

**Nota:** El operador MULT puede abreviarse con un asterisco (\*).

Este macro está disponible en IBM Interact.

## Ejemplos

$TEMP = 8 \text{ MULT } 4$ o $TEMP = 8 * 4$
Crea una nueva columna denominada TEMP que contiene el valor 32.
$TEMP = V1 * 8$
Crea una nueva columna denominada TEMP, donde cada valor es el contenido de la columna V1 multiplicado por ocho.
$TEMP = V1:V3 * 2$
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son dos veces el contenido de la columna V1, los valores de la columna VX son dos veces el contenido de la columna V2 y los valores de la columna VY son dos veces el contenido de la columna V3.
$TEMP = V1 * V1$
Crea una nueva columna denominada TEMP que contiene el cuadrado de cada valor en la columna V1.
$TEMP = V1 * V2$
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 multiplicado por el correspondiente valor de fila de la columna V2.
$TEMP = V1:V3 * V4:V6$
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores en V1 multiplicados por los valores de fila correspondientes de la columna V4. La columna VX multiplica la columnas V2 por V5. La columna VY multiplica la columna V3 por V6.

TEMP = V1[10:20] \* V2 o TEMP = V1[10:20] \* V2[1:11]

Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los valores de las filas 10 a 20 de la columna V1 multiplicados por los valores en las filas 1 a 11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
DIV	Divide un rango de datos especificado por otro
EXP	Calcula el número natural (e) elevado al contenido de cada celda del rango de datos especificado
POW	Calcula un valor base elevado a la(s) potencia(s) exponencial(es) especificada(s)

---

## NE

### Sintaxis

data1 NE data2 data1 != data2 data1 <> data2

### Parámetros

data1

El rango de celdas para comparar. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

data2

El/los número(s) con los que comparar todos los valores de la columna especificada. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en data2 debe ser igual al número de columnas en data1, a menos que data2 sea una constante. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

NE compara los dos rangos de datos especificados, devolviendo un uno si los valores no son iguales o un cero si son iguales. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene la correspondiente columna en data1 comparada con la columna correspondiente de data2 (es decir que la primera columna de data1 se compara con la primera columna de data, la segunda columna con la segunda columna, y a sí sucesivamente).

Si data2 es una constante, cada valor en data1 se compara con ese valor. Si data2 es una columna, los cálculos se realizan fila por fila. Los valores en data1 se

comparan con la primera fila de data2, la segunda fila con la segunda fila, y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** Utilizar una columna que contiene el mismo número  $x$  en cada fila como data2 es lo mismo que utilizar la constante  $x$  como data2.

**Nota:** El operador NE puede abreviarse con un signo de exclamación seguido de un signo de igual (!=) o por un signo menor que seguido de un signo de mayor que (<=).

Este macro está disponible en IBM Interact.

## Ejemplos

TEMP = 3 NE 4 o TEMP = 3 != 4 TEMP = 3 <= 4
Crea una nueva columna denominada TEMP que contiene el valor uno (dado que tres no es igual a cuatro).
TEMP = V1 != 8
Crea una nueva columna denominada TEMP, donde cada valor es uno si el correspondiente valor de fila de la columna V1 no es igual al número ocho. De contrario, el valor es cero.
TEMP = V1:V3 != 2
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son el contenido de la columna V1 comparados con el valor dos, los valores de la columna VX son el contenido de la columna V2 comparados con el valor dos y los valores de la columna VY son el contenido de la columna V3 comparados con el valor dos.
TEMP = V1 != V1
Crea una nueva columna denominada TEMP que contiene solo ceros(dado que todo número es igual a sí mismo).
TEMP = V1 != V2
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 comparado con el valor de fila correspondiente de la columna V2.
TEMP = V1:V3 != V4:V6
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores en V1 comparados con los valores de fila correspondientes de la columna V4. La columna VX compara las columnas V2 y V5. La columna VY compara las columnas V3 y V6.
TEMP = V1[10:20] != V2 o TEMP = V1[10:20] != V2[1:11]
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los resultados de la comparación de los valores en las filas 10 a 20 de la columna V1 con las filas 1 a 11 de las columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
EQ	Devuelve TRUE si un rango de datos es igual a otro
GE	Devuelve TRUE si un rango de datos es mayor o igual a otro

<b>Función</b>	<b>Descripción</b>
GT	Devuelve TRUE si un rango de datos es mayor que otro
LE	Devuelve TRUE si un rango de datos es menor o igual a otro
LT	Devuelve TRUE si un rango de datos es menor que otro

---

## **NORM\_MINMAX**



### **Sintaxis**

NORM\_MINMAX(data [, keyword]) NORM\_MINMAX(data, min, max [, keyword])  
 NORM\_MINMAX(data, base\_data [, keyword])

### **Parámetros**

data

Los valores numéricos para normalizar. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

min, max

Estos dos parámetros proporcionan un valor mínimo y un valor máximo a utilizar para la normalización. Deben ser constantes, excepto con la palabra clave ROW, en donde pueden ser constantes o columnas.

base\_data

Este parámetro especifica un rango de datos a utilizar para calcular el máximo y mínimo que se debe utilizar para la normalización. El número de columnas que se proporciona en base\_data debe ser el mismo número de columnas que el que se proporciona para data. Para obtener la definición de formato de base\_data (al igual que data), consulte la sección "Macro Function Parameters" en el capítulo de esta guía para el producto de IBM .

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las opciones siguientes:

ALL - Realiza el cálculo en todas las celdas de data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de data



Para obtener más detalles sobre la utilización de palabras clave en IBM Campaign, consulte “Especificaciones de formato” en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte “Especificaciones de formato” en la página 27.

## Descripción

NORM\_MINMAX calcula los valores normalizados del rango de datos especificado. Todos los valores devueltos estarán entre cero y uno, incluido. La normalización de mín/máx se realiza de la siguiente manera:

$$VX[y] = \frac{VX[y] - \min}{\max - \min}$$

donde *mín* y *máx* se determinan de la siguiente manera:

- Si *min* y *max* están proporcionados, estos valores se utilizan para el mínimo y máximo, respectivamente. Si estos parámetros se proporcionan con la palabra clave ROW, *min* y *max* pueden ser columnas, especificando un valor mínimo y máximo para cada fila de data. Si *min* y *max* son columnas, las columnas deben tener la misma longitud que data o escalar (es decir, contener un único valor que se utiliza como una constante que se aplica a todos los valores de la correspondiente columna de data).
- Si *base\_data* se proporciona, se calcula el mínimo y máximo de este rango de datos y se utiliza para normalizar data. Las columnas de *base\_data* deben contener dos o más valores de celda.
- Si no se proporciona ninguna de las opciones anteriores, que se excluyen mutuamente, los valores de mínimo y máximo se calculan automáticamente desde data.

**Nota:** Puesto que todos los valores devueltos están entre 0,0 y 1,0, los valores calculados utilizando Equation (anterior) que sean inferiores a 0,0 se devuelven como 0,0. Del mismo modo, los valores superiores a 1,0 se devuelven como 1,0. NORM\_MINMAX siempre devuelve un rango de datos con las mismas dimensiones que el rango de datos de entrada. La palabra clave ALL especifica que se debe calcular la desviación estándar y promedio sobre todo el rango de datos de entrada. La palabra clave COL especifica que se debe calcular una desviación estándar y promedio para cada columna de entrada y utilizar esos valores para normalizar esa columna. La palabra clave ROW especifica que se debe calcular una desviación estándar y promedio para cada fila del rango de datos especificado y utilizar esos valores para normalizar esa fila.

**Nota:** Si el mínimo y máximo son iguales, se devuelven ceros.

## Ejemplos

```
TEMP = NORM_MINMAX(3)
```

Crea una nueva columna denominada TEMP que contiene el valor cero.

<p>TEMP = NORM_MINMAX(COLUMN(3, 4, 5))</p> <p>Crea una nueva columna denominada TEMP que contiene los valores 0, 0.5 y 1. (El mínimo y máximo [3 y 5] se calculan desde el rango de datos de forma automática.)</p>
<p>TEMP = NORM_MINMAX(COLUMN(3, 4, 5), 0, 10)</p> <p>Crea una nueva columna denominada TEMP que contiene los valores 0.3, 0.4 y 0.5. (Esta vez, el mínimo y máximo [0 y 10] se proporcionan como argumentos.)</p>
<p>TEMP = NORM_MINMAX(V1)</p> <p>Crea una nueva columna denominada TEMP que contiene los valores normalizados del contenido de la columna V1. Los valores mínimos y máximos que se utilizan para la normalización se calculan sobre la columna V1.</p>
<p>TEMP = NORM_MINMAX(V1:V3)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. Los valores mínimos y máximos que se utilizan para la normalización se calculan sobre las columnas V1, V2 y V3.</p>
<p>TEMP = NORM_MINMAX(V1[1:5]:V3)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una con valores que se encuentran en las filas 1-5. El contenido de la columna TEMP son los valores normalizados de las filas correspondientes de la columna V1, el contenido de la columna VX son los valores normalizados de las filas correspondientes de la columna V2 y el contenido de la columna VY son los valores normalizados de las filas correspondientes de la columna V3. El mínimo y máximo, para fines de normalización, se calcula sobre las filas 1-5 de las columnas V1-V3.</p>
<p>TEMP = NORM_MINMAX(V1:V3, V4:V6)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. Los valores mínimos y máximos que se utilizan para la normalización se calculan sobre las columnas V4, V5 y V6.</p>
<p>TEMP = NORM_MINMAX(V1:V3, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. Los valores de mínimo y máximo que se utilizan para la normalización se calculan para cada columna independientemente (es decir, un mín/máx se calcula para la columna V1, un mín/máx independiente se calcula para la columna V2, etc.).</p>
<p>TEMP = NORM_MINMAX(V1[10:50]:V3, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una con valores que se encuentran en las primeras 41 filas. El contenido de la columna TEMP son los valores normalizados de las filas 10-50 de la columna V1, el contenido de la columna VX son los valores normalizados de las filas 10-50 de la columna V2 y el contenido de la columna VY son los valores normalizados de las filas 10-50 de la columna V2. El mínimo y máximo, para fines de normalización, se calcula de forma independiente en las filas 10-50 para cada columna.</p>
<p>TEMP = NORM_MINMAX(V1:V3, V4:V6, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. Los valores de mínimo y máximo utilizados para la normalización se calculan para cada columna, de forma independiente, utilizando las columnas V4-V6 (es decir, un mín/máx se calcula sobre la columna V4 para normalizar la columna V1, un mín/máx independiente se calcula sobre la columna V5 para normalizar la columna V2, etc.).</p>

<p>TEMP = NORM_MINMAX(V1:V3, ROW)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. Los valores mínimos y máximos que se utilizan para la normalización se calculan, de forma independiente, sobre cada fila de las columnas V1, V2 y V3.</p>
<p>TEMP = NORM_MINMAX(V1[10:20]:V3, ROW)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una con valores que se encuentran en las primeras 11 filas. El contenido de la columna TEMP son los valores normalizados de las filas 10-20 de la columna V1, el contenido de la columna VX son los valores normalizados de las filas 10-20 de la columna V2 y el contenido de la columna VY son los valores normalizados de las filas 10-20 de la columna V3. El mínimo y máximo, para fines de normalización, se calcula sobre cada una de las filas 10-20 de las columnas V1-V3.</p>
<p>TEMP = NORM_MINMAX(V1:V3, V8:V10, ROW)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. Los valores mínimos y máximos que se utilizan para la normalización se calculan independientemente para cada fila de las columnas V8-V10.</p>

### Funciones relacionadas

Función	Descripción
NORM_SIGMOID	Calcula la normalización sigmooidal de un rango de datos
NORM_ZSCORE	Calcula la normalización z-score de un rango de datos

---

## NORM\_SIGMOID



### Sintaxis

NORM\_SIGMOID(data [, keyword]) NORM\_SIGMOID(data, mean, std [, keyword])  
 NORM\_SIGMOID(data, base\_data [, keyword])

### Parámetros

data

Los valores para normalizar. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúe a cualquiera de los anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

mean, std

Estos dos parámetros proporcionan la desviación estándar y promedio a utilizar para la normalización. Deben ser constantes, excepto con la palabra clave ROW, en donde pueden ser constantes o columnas.

base\_data

Este parámetro especifica un rango de datos a utilizar para calcular la desviación estándar y promedio a utilizar para la normalización de data.

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las opciones siguientes:

ALL - Realiza el cálculo en todas las celdas de data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de data

Para obtener más detalles sobre la utilización de las palabras claves en IBM Campaign, consulte “Especificaciones de formato” en la página 9.

Para obtener más detalles sobre la utilización de palabrasclave en IBM PredictiveInsight, consulte “Especificaciones de formato” en la página 27.

## Descripción

NORM\_SIGMOID calcula los valores normalizados del rango de datos especificado. Una normalización sigmoideal redistribuye datos a lo largo de una curva sigmoideal, devolviendo valores entre -1,0 y +1,0, incluido. Básicamente, todos los datos dentro de una desviación estándar del promedio están distribuidos de forma lineal en el rango medio del sigmoide. Los valores atípicos están representados en las colas del sigmoide. Esto le permite conservar puntos de datos de valores atípicos muy grandes sin tener que sacrificar la capacidad de discriminación entre los puntos que sean cercanos al promedio.

La normalización sigmoideal se realiza de la siguiente manera:

$$VX[y] = \frac{1 - e^{-\alpha}}{1 + e^{-\alpha}}$$

donde

$$\alpha = \frac{VX[y] - \text{mean}}{\text{std}}$$

y *promedio* y *std* se determinan de la siguiente manera:

- Si *mean* y *std* se proporcionan, estos valores se utilizan para la desviación promedio y estándar, respectivamente. Si se proporciona estos parámetros con la palabra clave ROW, *mean* y *std* pueden ser columnas, especificando una desviación estándar y promedio para cada fila de data. Si *min* y *max* son columnas, las columnas deben tener la misma longitud que *data* o *escalar* (es decir, contener

un único valor que se utiliza como una constante que se aplica a todos los valores de la correspondiente columna de data).

- Si se proporciona base\_data, se calculan la desviación promedio y estándar de este rango de datos y se utilizan para normalizar data. Las columnas de base\_data deben contener dos o más valores de celda.
- Si no se proporciona ninguna de las anteriores opciones mutuamente excluyentes, la desviación promedio y estándar se calcula automáticamente a partir de data.

NORM\_SIGMOID siempre devuelve un rango de datos con las mismas dimensiones que el rango de datos de entrada. La palabra clave ALL especifica que se debe calcular la desviación estándar y promedio sobre todo el rango de datos de entrada. La palabra clave COL especifica que se debe calcular una desviación estándar y promedio para cada columna de entrada y utilizar esos valores para normalizar esa columna. La palabra clave ROW especifica que se debe calcular una desviación estándar y promedio para cada fila del rango de datos especificado y utilizar esos valores para normalizar esa fila.

**Nota:** Si la desviación estándar es cero, se devuelven ceros.

**Nota:** Para normalizar los datos utilizando el mismo rango base\_data (por ejemplo, en funciones de usuario recortadas), haga de mean y std constantes (esto puede realizarse utilizando la función de macro CONSTANT).

## Ejemplos

TEMP = NORM_SIGMOID(COLUMN(3, 4, 5))
Crea una nueva columna denominada TEMP que contiene los valores -0,55, 0 y 0,55. (La desviación promedio y estándar [4 y 0.816] se calculan automáticamente a partir del rango de datos).
TEMP = NORM_SIGMOID(COLUMN(3, 4, 5), 3.5, 1.2)
Crea una nueva columna denominada TEMP que contiene los valores -0,21, 0,21 y 0,55. (En esta ocasión la desviación promedio y estándar [3.5 y 1.2] se proporcionan como argumentos).
TEMP = NORM_SIGMOID(V1) o TEMP = NORM_SIGMOID(V1,ALL)
Crea una nueva columna denominada TEMP que contiene los valores normalizados del contenido de la columna V1. La desviación estándar y promedio que se utiliza para la normalización se calcula sobre la columna V1.
TEMP = NORM_SIGMOID(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. La desviación estándar y promedio que se utiliza para la normalización se calcula sobre las columnas V1, V2 y V3.
TEMP = NORM_SIGMOID(V1[10:50]:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una con valores en las filas 1 a 41. El contenido de la columna TEMP es los valores normalizados de las filas 10 a 50 de la columna V1, el contenido de la columna VX es los valores normalizados de las filas 10 a 50 de la columna V2 y el contenido de la columna VY es los valores normalizados de las filas 10 a 50 de la columna V3. La desviación promedio y estándar para la normalización se calculan sobre las filas 10 a 50 de las columnas V1-V3.

<p>TEMP = NORM_SIGMOID(V1:V3, V4)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. La desviación estándar y promedio que se utiliza para la normalización se calcula sobre la columna V4.</p>
<p>TEMP = NORM_SIGMOID(V1:V3, V4:V8)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. La desviación estándar y promedio que se utiliza para la normalización se calcula sobre las columnas V4-V8.</p>
<p>TEMP = NORM_SIGMOID(V1:V3, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. La desviación promedio y estándar utilizada para la normalización se calcula para cada columna independientemente, (es decir, se calcula una desviación estándar y promedio para la columna V1, se calcula otra desviación estándar y promedio para la columna V2, etc.).</p>
<p>TEMP = NORM_SIGMOID(V1[10:50]:V3, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una con valores en las filas 1 a 41. El contenido de la columna TEMP es los valores normalizados de las filas 10 a 50 de la columna V1, el contenido de la columna VX es los valores normalizados de las filas 10 a 50 de la columna V2 y el contenido de la columna VY es los valores normalizados de las filas 10 a 50 de la columna V3. La desviación promedio y estándar para la normalización se calculan sobre las filas 10 a 50 de las columnas V1-V3. La desviación promedio y estándar para la normalización se calculan independientemente para cada columna.</p>
<p>TEMP = NORM_SIGMOID(V1:V3, V4:V6, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. La desviación estándar y promedio utilizada para la normalización se calcula para cada columna, de forma independiente, utilizando las columnas V4-V6 (es decir, una desviación estándar y promedio se calcula sobre la columna V4 para normalizar la columna V1, una desviación estándar y promedio independiente se calcula sobre la columna V5 para normalizar la columna V2, etc.).</p>
<p>TEMP = NORM_SIGMOID(V1:V3, ROW)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. La desviación promedio y estándar utilizada para la normalización se calcula independientemente sobre cada fila de las columnas V1, V2 y V3.</p>
<p>TEMP = NORM_SIGMOID(V1[10:50]:V3, ROW)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una con valores en las filas 1 a 41. El contenido de la columna TEMP es los valores normalizados de las filas 10 a 50 de la columna V1, el contenido de la columna VX es los valores normalizados de las filas 10 a 50 de la columna V2 y el contenido de la columna VY es los valores normalizados de las filas 10 a 50 de la columna V3. La desviación promedio y estándar para la normalización se calculan sobre las filas 10 a 50 de las columnas V1-V3. La desviación promedio y estándar para la normalización se calculan sobre cada fila de las columnas V1-V3.</p>
<p>TEMP = NORM_SIGMOID(V1:V3, V4:V10, ROW)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. La desviación promedio y estándar utilizada para la normalización se calcula independientemente para cada fila de las columnas V4-V10.</p>

## Funciones relacionadas

Función	Descripción
NORM_MINMAX	Calcula la normalización min/max de un rango de datos
NORM_ZSCORE	Calcula la normalización z-score de un rango de datos

---

## NORM\_ZSCORE



### Sintaxis

```
NORM_ZSCORE(data [, keyword]) NORM_ZSCORE(data, mean, std [, keyword])  
NORM_ZSCORE(data, base_data [, keyword])
```

### Parámetros

data

Los valores numéricos para normalizar. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

mean, std

Estos dos parámetros proporcionan la desviación promedio y estándar a utilizar para la normalización. Deben ser constantes, excepto con la palabra clave ROW, en donde pueden ser constantes o columnas.

base\_data

Este parámetro especifica un rango de datos a utilizar para calcular la desviación estándar y promedio a utilizar para la normalización de data.

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo sobre el rango de datos de entrada especificado. Seleccione una de las siguientes opciones:

ALL - Realiza el cálculo en todas las celdas de data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de data

Para obtener más detalles sobre la utilización de las palabras claves en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

## Descripción

NORM\_ZSCORE calcula los valores normalizados del rango de datos especificado. La normalización de la puntuación  $z$  se realiza de la siguiente manera:

$$VX[y] = \frac{VX[y] - \text{mean}}{\text{std}}$$

dónde *mean* y *std* se determinan de la siguiente manera:

- Si *mean* y *std* se proporcionan, estos valores se utilizan para la desviación promedio y estándar, respectivamente. Si se proporciona estos parámetros con la palabra clave ROW, *mean* y *std* pueden ser columnas, especificando una desviación estándar y promedio para cada fila de data. Si *min* y *max* son columnas, las columnas deben tener la misma longitud que *data* o *escalar* (es decir, contener un único valor que se utiliza como una constante que se aplica a todos los valores de la correspondiente columna de data).
- Si se proporciona *base\_data*, se calculan la desviación promedio y estándar de este rango de datos y se utilizan para normalizar *data*. Las columnas de *base\_data* deben contener dos o más valores de celda.
- Si no se proporciona ninguna de las anteriores opciones mutuamente excluyentes, la desviación promedio y estándar se calcula automáticamente a partir de *data*.

NORM\_ZSCORE siempre devuelve un rango de datos con las mismas dimensiones que el rango de datos de entrada. Calcula una desviación estándar y promedio por cada columna de entrada y utiliza esos valores para normalizar la columna.

**Nota:** Si la desviación estándar es cero, se devuelve todo ceros.

## Ejemplos

```
TEMP = NORM_ZSCORE(COLUMN(3, 4, 5))
```

Crea una nueva columna denominada TEMP que contiene los valores -1.22, 0 y 1.22. (La desviación promedio y estándar [4 y 0.816] se calculan automáticamente a partir del rango de datos).

```
TEMP = NORM_ZSCORE(COLUMN(3, 4, 5), 3.5, 1.2)
```

Crea una nueva columna denominada TEMP que contiene los valores -0.42, 0.42 y 1.25. (En esta ocasión la desviación promedio y estándar [3.5 y 1.2] se proporcionan como argumentos).

```
TEMP = NORM_ZSCORE(V1)
```

Crea una nueva columna denominada TEMP que contiene los valores normalizados del contenido de la columna V1. La desviación promedio y estándar utilizada para la normalización se calcula sobre la columna V1.

```
TEMP = NORM_ZSCORE(V1:V3)
```

Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. La desviación promedio y estándar utilizada para la normalización se calcula para cada columna independientemente, (es decir, se calculan una desviación estándar y promedio para la columna V1, se calculan otra desviación estándar y promedio para la columna V2, etc.).



<p>TEMP = NORM_ZSCORE(V1[10:50]:V3)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una con valores en las filas 1 a 41. El contenido de la columna TEMP es los valores normalizados de las filas 10 a 50 de la columna V1, el contenido de la columna VX es los valores normalizados de las filas 10 a 50 de la columna V2 y el contenido de la columna VY es los valores normalizados de las filas 10 a 50 de la columna V3. La desviación promedio y estándar para la normalización se calculan sobre las filas 10 a 50 de las columnas V1-V3. La desviación promedio y estándar para la normalización se calculan independientemente para cada columna.</p>
<p>TEMP = NORM_ZSCORE(V1:V3, V4:V6)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. La desviación estándar y promedio utilizada para la normalización se calcula para cada columna, de forma independiente, utilizando las columnas V4-V6 (es decir, una desviación estándar y promedio se calcula sobre la columna V4 para normalizar la columna V1, una desviación estándar y promedio independiente se calcula sobre la columna V5 para normalizar la columna V2, etc.).</p>
<p>TEMP = NORM_ZSCORE(V1:V3, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. La desviación promedio y estándar utilizada para la normalización se calcula para cada columna independientemente, (es decir, se calculan una desviación estándar y promedio para la columna V1, se calculan otra desviación estándar y promedio para la columna V2, etc.).</p>
<p>TEMP = NORM_ZSCORE(V1[10:50]:V3, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una con valores en las filas 1 a 41. El contenido de la columna TEMP es los valores normalizados de las filas 10 a 50 de la columna V1, el contenido de la columna VX es los valores normalizados de las filas 10 a 50 de la columna V2 y el contenido de la columna VY es los valores normalizados de las filas 10 a 50 de la columna V3. La desviación promedio y estándar para la normalización se calculan sobre las filas 10 a 50 de las columnas V1-V3. La desviación promedio y estándar para la normalización se calculan independientemente para cada columna.</p>
<p>TEMP = NORM_ZSCORE(V1[10:50]:V3, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. La desviación estándar y promedio utilizada para la normalización se calcula para cada columna, de forma independiente, utilizando las columnas V4-V6 (es decir, una desviación estándar y promedio se calcula sobre la columna V4 para normalizar la columna V1, una desviación estándar y promedio independiente se calcula sobre la columna V5 para normalizar la columna V2, etc.).</p>
<p>TEMP = NORM_ZSCORE (V1:V3, ROW)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. La desviación promedio y estándar utilizada para la normalización se calcula independientemente sobre cada fila de las columnas V1, V2 y V3.</p>
<p>TEMP = NORM_ZSCORE(V1[10:50]:V3, ROW)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una con valores en las filas 1 a 41. El contenido de la columna TEMP es los valores normalizados de las filas 10 a 50 de la columna V1, el contenido de la columna VX es los valores normalizados de las filas 10 a 50 de la columna V2 y el contenido de la columna VY es los valores normalizados de las filas 10 a 50 de la columna V3. La desviación promedio y estándar para la normalización se calculan sobre las filas 10 a 50 de las columnas V1-V3. La desviación promedio y estándar para la normalización se calculan sobre cada fila de las columnas V1-V3.</p>

TEMP = NORM\_ZSCORE(V1:V3, V4:V10, ROW)

Crea tres nuevas columnas denominadas TEMP, VX y VY. Cada una contiene los valores normalizados del contenido de las columnas V1, V2 y V3, respectivamente. La desviación promedio y estándar utilizada para la normalización se calcula independientemente para cada fila de las columnas V4-V10.

## Funciones relacionadas

Función	Descripción
NORM_MINMAX	Calcula la normalización min/max de un rango de datos
NORM_SIGMOID	Calcula la normalización sigmoideal de un rango de datos

---

## NOT

### Sintaxis

NOT(data) ! data

### Parámetros

data

Los valores numéricos a los que calcular el NOT lógico. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

NOT devuelve el NOT lógico de los valores en el rango de datos especificado. Devuelve una columna nueva por cada columna de entrada, cada una de las cuales el NOT lógico de los valores en la correspondiente columna de entrada. Esta función devuelve cero para los valores distintos de cero y uno para los valores cero.

**Nota:** El operador NOT puede abreviarse con un signo de exclamación (!). Utilice el signo de exclamación antes del valor de datos (por ejemplo, para especificar NOT(V1), puede simplemente escribir !V1).

Este macro está disponible en IBM Interact.

### Ejemplos

TEMP = NOT(3.2) o TEMP = !1

Crea una nueva columna denominada TEMP que contiene el valor cero.

TEMP = !0 o TEMP = !(2+2=3)

Crea una nueva columna denominada TEMP que contiene el valor uno.

TEMP = !V1
Crea una nueva columna denominada TEMP, donde cada valor es el NOT lógico de los valores en la columna V1.
TEMP = !V1:V3
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores de la columna TEMP son los NOT lógicos de los valores en la columna V1, los valores de la columna VX son los NOT lógicos de los valores en la columna V2 y los valores de la columna VY son los NOT lógicos de los valores en la columna V3.
TEMP = !V1[10:20]
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los NOT lógicos de los valores en las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = !V1[1:5]:V2
Crea dos nuevas columnas denominadas TEMP y VX cada una con valores en filas 1-5 (las otras celdas están vacías). Los valores en la columna TEMP son los NOT lógicos de los valores de las correspondientes filas de la columna V1, y los valores en la columna VX son los NOT lógicos de los valores de las correspondientes filas de la columna V2.

## Funciones relacionadas

Función	Descripción
AND	Calcula el número NOT lógico del contenido del rango de datos especificado AND lógico entre dos rangos de datos especificados
INVERSE	Calcula el negativo del contenido del rango de datos especificado
OR	Calcula el OR lógico entre dos rangos de datos especificados
SIGN	Calcula el signo (positivo o negativo) de los valores del rango de datos especificado

---

## NPV



### Sintaxis

NPV(data, rate, [, keyword])

### Parámetros

data

Los valores numéricos que representan el flujo neto de caja esperado utilizado para calcular el valor neto actual. Este puede ser una fila, una columna, un rango de celdas o una expresión que evalúa cualquiera de los elementos anteriores.

rate

El valor numérico que representa la tasa de descuento durante la longitud de un periodo.

keyword

Esta palabra clave determina cómo se realiza el cálculo sobre el rango de datos de entrada. Si no proporciona la palabra clave, ROW se utiliza como valor predeterminado. Seleccione una de las siguientes opciones:

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de datos

## Descripción

NPV calcula el valor neto actual de una inversión basándose en una serie de flujos de efectivo periódicos y una tasa de descuento. El valor neto actual de una inversión es el valor de hoy de una serie de pagos futuros (valores negativos) e ingresos (valores positivos). NPV se calcula mediante la fórmula:

$$NPV = \sum_{t=1}^n \frac{\text{data}_t}{(1 + \text{rate})^t}$$

donde  $n$  es el número de flujos de efectivo (valores de datos).

El orden de los valores de datos se utiliza para interpretar el orden de los flujos de efectivo. La inversión NPV empieza un periodo antes de la fecha del primer valor de flujo de efectivo y termina con el último flujo de caja en la lista.

**Nota:** Si el primer flujo de efectivo se produce al principio del primer periodo, el primer valor debe ser añadido al resultado de la NPV, que no está incluido en los valores de datos.

El número de columnas devueltas por NPV depende de keyword.

- Si keyword es COL, NPV devuelve una nueva columna para cada columna de entrada. Cada nueva columna contiene un valor (el valor neto actual de todas las celdas en la columna de entrada correspondiente).
- Si keyword es ROW, NPV devuelve una nueva columna que contiene el valor neto actual en todas las filas de data.

**Nota:** Las celdas en blanco se ignoran en el NPV.

## Ejemplos

`TEMP = NPV(V1:V3,.10)` o `TEMP = NPV(V1:V3,.10, ROW)`

Crea una nueva columna denominada TEMP que contiene un valor único que es el NPV de los valores de las columnas V1, V2 y V3. La tasa de descuento que se utiliza es 10%.

TEMP = NPV(V1, .10, COL)

Crea una nueva columna denominada TEMP que contiene un valor único que es el NPV de los contenidos de la columna V1 que utiliza una tasa de descuento de 10%.

TEMP = NPV(V1:V3, .10) - 1000

Crea una nueva columna denominada TEMP que contiene un valor único que es el NPV de los contenidos de las columnas V1, V2 y V3 con un pago inicial de 1000. La tasa de descuento es 10%.

TEMP = NPV(V1[10:20], .10L, COL)

Crea una nueva columna denominada TEMP que contiene un valor único que es la NPV de las celdas de las filas 10-20 de la columna V1. La tasa de descuento es 10%.

TEMP = NPV(V1[1:5]:V4, .10)

Crea una nueva columna denominada TEMP que contiene un valor único que es la NPV de las celdas de las filas 1-5 de las columnas V1 a V4.

## NUMBER

### Sintaxis

NUMBER(data [, conversion\_keyword])

### Parámetros

data

Los datos de texto ASCII a convertir a valores numéricos. Este puede ser un texto ASCII en comillas, una columna de texto, un rango de celdas que contiene texto o una expresión que evalúa a algunos de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

conversion\_keyword

Esta palabra clave opcional especifica cómo interpretar formatos de texto para fechas y horas. Seleccione una de las palabras claves en la siguiente tabla.

**Nota:** Si no especifica este parámetro, el valor predeterminado es 1.

La conversión de la palabra clave	Formato	Descripción
0	#####	Convierte los 5 primeros caracteres de cada serie de texto en un número exclusivo
1	\$ ( default)	Convierte valores de dólar a numéricos (por ejemplo, "\$123.45" a 123.45)
2	%	Convierte un valor de porcentaje a numéricos (por ejemplo, "50%" a 0.5)

3	dd/mm/aa hh:mm	Convierte una fecha y una hora para el número de días transcurridos desde el 1 de enero, 0000 (1900 es automáticamente añadido al año aa)
4	dd-mm-aa	Convierte una fecha para el número de días transcurridos desde el 1 de enero, 0000 (1900 es automáticamente añadido al año aa)
5	dd/mm/aa	Convierte una fecha para el número de días transcurridos desde el 1 de enero, 0000 (1900 es automáticamente añadido al año aa)
6	mm-aa	Convierte una fecha para un número de días transcurridos desde el primer día del mes especificado y el 1 de enero, 0000 (1900 es automáticamente añadido al año aa)
7	dd-mm	Convierte una fecha en el número de días desde el principio del año (por ejemplo, "01-FEB" a32)
8	mmm	Convierte una abreviatura de mes de tres letras a un valor entre 1-12 (por ejemplo, "DIC" a 12)
9	{Enero   Febrero   Marzo ... }	Convierte un nombre de mes completamente explicado a un valor entre 1-12 (por ejemplo, "Marzo" a 3)
10	{Dom   Lun   Mar ... }	Convierte una abreviación de tres días de la semana a un valor entre 0-6 donde el domingo marca el principio de la semana (por ejemplo, "Dom" a 0)
11	{Domingo   Lunes   Martes ... }	Convierte un nombre de día de semana completamente explicado a un valor entre 0-6, donde el domingo marca el inicio de la semana (por ejemplo, "Lunes" a 1)
12	hh:mm:ss {AM   PM}	Convierte la hora para el número de segundos transcurridos desde 00:00:00 AM (medianoche) (por ejemplo, "01:00:00 AM" a 3600)

13	hh:mm:ss	Convierte la hora para el número de segundos transcurridos desde 00:00:00 AM (medianoche) (por ejemplo, "01:00:00" a 3600)
14	hh:mm {AM   PM}	Convierte la hora para el número de minutos transcurridos desde 00:00:00 AM (medianoche) (por ejemplo, "01:00 AM" a 60)
15	hh:mm	Convierte la hora para el número de minutos transcurridos desde 00:00:00 AM (medianoche) (por ejemplo, "01:00" a 60)
16	mm:ss	Convierte la hora para el número de segundos transcurridos desde 00:00:00 AM (medianoche) (por ejemplo, "30:00" a 1800)
17	ddmm	Convierte una fecha en el número de días desde el principio del año (por ejemplo, "3101" a 31)
18	ddmmm	Convierte una fecha en el número de días desde el principio del año (por ejemplo "31ENE" a 31)
19	ddmmaa	Convierte una fecha para el número de días transcurridos desde el 1 de enero, 0000 (1900 es automáticamente añadido al año si aa es menor o igual a 20; de lo contrario se añade 2000)
20	ddmmaaaa	Convierte una fecha en el número de días transcurridos desde el 1 de enero, 0000 (por ejemplo, "31ENE0000" a 31)
21	ddmmaa	Convierte una fecha para el número de días transcurridos desde el 1 de enero, 0000 (1900 es automáticamente añadido al año si aa es menor o igual a 20; de lo contrario se añade 2000)
22	ddmmaaaa	Convierte una fecha en el número de días transcurridos desde el 1 de enero, 0000 (por ejemplo, "31010000" a 31)
23	mmdd	Convierte una fecha en el número de días desde el principio del año (por ejemplo, "0131" a 31)

24	mmddaa	Convierte una fecha para el número de días transcurridos desde el 1 de enero, 0000 (1900 es automáticamente añadido al año si aa es menor o igual a 20; de lo contrario se añade 2000)
25	mmddaaaa	Convierte una fecha en el número de días transcurridos desde el 1 de enero, 0000 (por ejemplo, "01010001" a 366)
26	mmm	Convierte una abreviatura de mes de tres letras a un valor entre 1-12 (por ejemplo, "MAR" a 3) [Note que esta es igual a la conversión de la palabra clave 8]
27	mmdd	Convierte una fecha en el número de días desde el principio del año (por ejemplo, "ENE31" a 31)
28	mmddaa	Convierte una fecha para el número de días transcurridos desde el 1 de enero, 0000 (1900 es automáticamente añadido al año si aa es menor o igual a 20; de lo contrario se añade 2000)
29	mmddaaaa	Convierte una fecha para el número de días transcurridos desde el 1 de enero, 0000 (por ejemplo, "FEB010001" a 32)
30	mmaaa	Convierte una fecha para el número de días transcurridos desde el primer día del mes especificado y el 1 de enero, 0000 (1900 es automáticamente añadido al año si aa es menor o igual a 20; de lo contrario se añade 2000)
31	mmaaaa	Convierte una fecha para el número de días transcurridos entre el primer día del mes especificado y el 1 de enero, 0000 (por ejemplo, "FEB0001" a 32)



32	mmaa	Convierte una fecha para el número de días transcurridos desde el primer día del mes especificado y el 1 de enero, 0000 (1900 es automáticamente añadido al año si aa es menor o igual a 20; de lo contrario se añade 2000)
33	mmaaaa	Convierte una fecha para el número de días transcurridos entre el primer día del mes especificado y el 1 de enero, 0000 (por ejemplo, "020001" a 32)
34	aamm	Convierte una fecha para el número de días transcurridos desde el primer día del mes especificado y el 1 de enero, 0000 (1900 es automáticamente añadido al año si aa es menor o igual a 20; de lo contrario se añade 2000)
35	aamddd	Convierte una fecha para el número de días transcurridos desde el 1 de enero, 0000 (1900 es automáticamente añadido al año si aa es menor o igual a 20; de lo contrario se añade 2000)
36	aammm	Convierte una fecha para el número de días transcurridos desde el primer día del mes especificado y el 1 de enero, 0000 (1900 es automáticamente añadido al año si aa es menor o igual a 20; de lo contrario se añade 2000)
37	aamddd	Convierte una fecha para el número de días transcurridos desde el 1 de enero, 0000 (1900 es automáticamente añadido al año si aa es menor o igual a 20; de lo contrario se añade 2000)
38	aaaa	Convierte el año el número de años transcurridos desde el año 0000 (por ejemplo, "1998" a 1998)
39	aaaamm	Convierte una fecha para el número de días transcurridos entre el primer día del mes especificado y el 1 de enero, 0000 (por ejemplo, "000102" a 32)

40	aaaamdd	Convierte una fecha para el número de días transcurridos desde el 1 de enero, 0000 (por ejemplo, "00010201" a 32)
41	aaaamm	Convierte una fecha para el número de días transcurridos entre el primer día del mes especificado y el 1 de enero, 0000 (por ejemplo, "000102" a 32)
42	aaaamddd	Convierte una fecha para el número de días transcurridos desde el 1 de enero, 0000 (por ejemplo, "0001FEB01" a 32)
43	<day>* <month>	Convierte cualquier fecha delimitada con días seguido del mes para los números de días transcurridos desde el comienzo del año (por ejemplo, "15-ENE" a 15)
44	<day>* <month>* <year>	Convierte cualquier fecha delimitada con días que aparecen antes del mes seguido del año al número de días transcurridos desde el 1 de enero, 0000 (por ejemplo, "1/1/0001" a 366)
45	<month>* <day>	Convierte cualquier fecha delimitada con el mes seguido del día en números de días desde el comienzo del año (por ejemplo, "ENE 31" a 31)
46	<month>* <day>* <year>	Convierte cualquier fecha delimitada con el mes seguido del día y del año en el número de días transcurridos desde el 1 de enero, 0000 (por ejemplo, "ENE 1, 0001" a 366)
47	<month>* <year>	Convierte cualquier fecha delimitada con el mes seguido del año en el número de días transcurridos entre el primer día del mes especificado y el 1 de enero, 0000
48	<year>* <month>	Convierte cualquier fecha delimitada con el año seguido del mes en el número de días transcurridos entre el primero del mes especificado y el 1 de enero, 0000

49	<year>* <month>* <day>	Convierte cualquier fecha delimitada con el mes seguido del día y del año en el número de días transcurridos desde el 1 de enero, 0000 (por ejemplo, "0001/01/01" a 366)
50	aa	Convierte el año en el número de años transcurridos desde el año 0000 (por ejemplo, "97" a 97)
51	mm	Convierte el mes en un valor entre 1-12 (por ejemplo, "SEP" a 9)
52	dd	Convierte el día en un valor entre 1-31 (por ejemplo, "28" a 28)
53	{Enero   Febrero   Marzo ... }	Convierte un nombre de mes completamente explicado a un valor entre 1-12 (por ejemplo, "Marzo" a 3) [Notar que esta es la misma conversión de la palabra clave 9]
54	{Domingo   Lunes   Martes ... }	Convierte un nombre de día de semana completamente detallado en un valor entre 1-7, donde el domingo marca el inicio de la semana (por ejemplo, "Domingo" a 1)
55	{Dom   Lun   Mar ... }	Convierte una abreviación de tres días de días de semana en un valor entre 1-7, donde el domingo marca el inicio de la semana (por ejemplo, "Dom" a 1)

## Descripción

NÚMERO convierte los valores de texto en el rango de datos especificado en valores numéricos que utilizan el formato específico para convertir fechas y horas. Si una cadena de texto no se puede analizar con la utilización de conversión\_palabra clave, NÚMERO generará un error. Formato 0 convierte los cinco primeros caracteres de cada serie de texto en diferentes números para cada cadena de texto exclusiva. Esta es una forma fácil de cambiar una columna de texto en clases exclusivas de salidas a un clasificador.

Los formatos delimitados(conversión de palabras 43-49) soporta cualquiera de los siguientes como delimitadores:

- / (barra inclinada)
- - (guión)
- , (coma)
- " " (espacio)
- : (dos puntos)

Los meses se pueden representar como mm o mmm; los días se pueden representar como d o dd; los años se pueden representar como aa o aaaa.

**Nota:** En el soporte del año 2000 de conformidad, todos los años en fechas pueden ser designados como aaaa en lugar de aa. Para la compatibilidad, la conversión de palabras clave 1-16, aa (2-dígitos años) tiene el 1900 añadido automáticamente. Para la conversión de palabras claves 17-55, aa < umbral tiene 2000 añadido automáticamente; aa ≥ umbral tiene 1900 añadido automáticamente.

**Nota:** El año 2000 umbral valor se establece en la pestaña **datos de limpieza** de la ventana **Configuración avanzada** (invocar utilizando **Opciones > Valores > Configuración avanzada**).

**Nota:** Si cambia el valor del año 2000 umbral de valor, debe actualizar todas las funciones de macro utilizando la función de macro NÚMERO para manipular los valores de fecha de 2 dígitos años. Para forzar una actualización de una función de macro, puede hacer cualquier edición (por ejemplo, añadir un espacio y suprimiéndolo) y pulsando el icono de marca de selección para aceptar el cambio.

**Nota:** Cuando se utiliza el formato 0, solo los cinco primeros caracteres de cada serie de texto se utilizan para generar un número exclusivo. Todas las series con los mismos primeros cinco caracteres se convertirán en el mismo valor numérico. La misma serie de texto producirá el mismo valor numérico cada vez, incluso entre distintas hojas. Si es necesario, utilice series de macros para manipular series de modo que los cinco primeros caracteres exclusivamente definan una clase. Tenga en cuenta que los valores numéricos resultantes pueden ser muy pequeños. Utilice la ventana **Visualizar formatos** para aumentar el número de posiciones decimales visualizadas, o cambie el formato al modo exponencial ( 00E+00).

Este macro está disponible en IBM Interact.

## Ejemplos

```
TEMP = NUMBER("$1.23") o TEMP = NUMBER("123%", 2)
```

Crea una nueva columna denominada TEMP que contiene el número 1.23.

```
TEMP = NUMBER(column("Jan", "Mar", "Dec", 8))
```

Crea una nueva columna denominada TEMP que contiene los números 1, 3 y 12.

```
TEMP = NUMBER("1:52 PM", 14)
```

Crea una nueva columna denominada TEMP que contiene el número 832.

```
TEMP = NUMBER("1/1/95", 5)
```

Crea una nueva columna denominada TEMP que contiene el número 728660.

```
TEMP = NUMBER(V1)
```

Crea una nueva columna denominada TEMP que contiene los valores numéricos de la cadena de texto en la columna V1. Los valores de dólar se convierten en valores numéricos correctamente. ??? 's ha devuelto para series de texto que no se pueden analizar utilizando el formato \$.

TEMP = NUMBER(V1:V3, 4)

Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores numéricos de las cadenas de texto en la columna V1. La columna VX contiene los valores numéricos de las cadenas de texto en la columna V2. La columna VY contiene los valores numéricos de las cadenas de texto en la columna V3. Todas las fechas en el formato dd-aaa-aa se convierten en el número de días desplazados desde el 1 de enero, 0000. ??? 's son devueltos para series de texto que no se pueden analizar utilizando el formato \$.

TEMP = NUMBER(V1[10:20]:V2, 10)

Crea dos nuevas columnas denominadas TEMP y VX. La columna TEMP contiene los valores numéricos de las cadenas de texto en las filas 10-20 de la columna V1. La columna VX contiene los valores numéricos de las cadenas de texto en las filas 10-20 de la columna V2. Todas las representaciones de caracteres de días de la semana se convierten en los números 0-6 (0 = Domingo, 6= Sábado). Si no hay ninguna coincidencia para un nombre de día de la semana, ??? se devuelve.

TEMP = NUMBER(V1, 0)

Suponiendo que esa columna V1 contiene todas las series de texto de cinco dígitos, crea una nueva columna denominada TEMP que contiene un valor numérico diferente para cada serie exclusiva.

## Funciones relacionadas

Función	Descripción
WEEKDAY	Convierte cadenas de fecha de texto ASCII en el día de la semana

---

## OFFSET



### Sintaxis

OFFSET(data)

### Parámetros

data

Los valores a los que calcular el desplazamiento. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

OFFSET devuelve los valores de desplazamiento del rango de datos especificado del primer valor. Devuelve una columna nueva que contiene los valores de desplazamiento, comienza con uno y continúa hasta la longitud de la columna más larga del rango de datos.

## Ejemplos

TEMP = OFFSET(5)

Crea una nueva columna denominada TEMP que contiene el valor uno.

TEMP = OFFSET(V1)

Crea una nueva columna denominada TEMP, donde cada valor es el desplazamiento de la columna V1, comenzando con uno y terminando con la longitud de la columna V1.

TEMP = OFFSET(V1:V3)

Crea una nueva columna denominada TEMP, donde cada valor es el desplazamiento, comenzando con uno y terminando con la columna más larga de V1, V2 o V3.

## Funciones relacionadas

Función	Descripción
COUNT	Cuenta la cantidad de celdas que contienen valores en el rango de datos especificado
DELAY	Devuelve la o las columnas de entrada retardadas por un número específico de pasos de tiempo

---

## OR

### Sintaxis

data1 OR data2 data1 || data2

### Parámetros

data1

Los números a los que aplicar el OR lógico con los valores en data2. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

data2

El o los números a los que aplicar el OR lógico con los valores en data1. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en data2 debe ser igual al número de columnas en data1, a menos que data2 sea una constante. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

OR calcula el OR lógico entre los dos rangos de datos especificados. Devuelve una columna nueva por cada columna de entrada, cada una de las cuales contiene el OR lógico entre la correspondiente columna en data1 y la correspondiente columna

de data2 (es decir, el OR lógico entre la primera columna de data1 y la primera columna de data, la segunda columna con la segunda columna, etc.).

Si data2 es una constante, se aplica el OR lógico de cada valor en data1 con ese valor. Si data2 contiene una o más columnas, los cálculos se realizan fila por fila entre una columna de data1 y una columna de data2. Se aplica el OR lógico entre la primera fila de data1 y el valor de la primera fila de data2, la segunda fila con la segunda fila, etc. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** Utilizar una columna que contiene el mismo número  $x$  en cada fila como data2 es lo mismo que utilizar la constante  $x$  como data2.

**Nota:** El operador OR puede ser abreviado con dos barras verticales (||). Utilice las dos barras verticales para separar los dos argumentos (por ejemplo, para especificar V1 OR 3, simplemente escriba V1||3).

Este macro está disponible en IBM Interact.

## Ejemplos

TEMP = 1 OR 8 o TEMP = 1    8
Crea una nueva columna denominada TEMP que contiene el valor uno (cualquier número distinto de cero es tratado como uno).
TEMP = V1    1
Crea una nueva columna denominada TEMP que contiene todo unos (el OR lógico entre cualquier valor y el número uno produce uno).
TEMP = V1    V2
Crea una nueva columna denominada TEMP, donde cada valor es el OR lógico entre el valor de la fila en la columna V1 y la fila correspondiente de la columna V2.
TEMP = V1:V3    V4:V6
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los OR lógicos entre los valores en V1 y los correspondientes valores de fila de la columna V4. La columna VX contiene los OR lógicos entre los valores de las columnas V2 y V5. La columna VY contiene los OR lógicos entre los valores de las columnas V3 y V6.
TEMP = V1[10:20]    V2
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen el resultado de aplicar el lógico OR a los valores de las filas 10 a 20 de las columnas V1 y V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
AND	Calcula el número NOT lógico del contenido del rango de datos especificado AND lógico entre dos rangos de datos especificados
NOT	Calcula el número NOT lógico del contenido del rango de datos rango de datos especificado

## PCA



### Sintaxis

PCA(data)

### Parámetros

data

Los valores numéricos para los que calcular los componentes principales. Estos pueden ser una constante, una columna, un rango de celdas o una expresión que evalúe a cualquiera de los anteriores.

### Descripción

PCA realiza un análisis de los componentes principales en el rango de datos especificado. Busca los autovectores ortogonales para el rango de datos especificado por de datos utilizando una descomposición de valores singular. Devuelve una columna nueva para cada una de las  $n$  columnas especificadas como entrada, más una columna adicional. Las primeras  $n$  columnas contienen los autovectores (cada autovector se lee como una fila en las  $n$  columnas). La última columna devuelta contiene las magnitudes correspondientes de valores propios. Los autovectores se ordenan de acuerdo con sus correspondientes valores propios.

**Nota:** Los valores que faltan (por ejemplo, celdas vacías y ???) se cuentan como ceros. Las columnas cortas de data se rellenan con ceros hasta alcanzar la longitud de la columna más larga.

A continuación se muestran los detalles de cómo se calcula PCA:

- Cada una de las  $k$  filas de data es un vector de  $n$  dimensiones  $v_i$  ( $n$  es el número de columnas en data). Estos se utilizan para calcular la matriz de correlación  $A$  del modo siguiente:

$$A = \sum_{i=1}^k v_i v_i^T$$

- La matriz de correlación  $A$  de  $n$  por  $n$  se descompone utilizando una descomposición de valores singular en tres matrices:

$$A = U \Sigma U^T$$

Las filas de  $U$  son los autovectores de  $A$  y  $\Sigma$  es una matriz diagonal en donde cada elemento diagonal es la magnitud de los valores propios de  $A$ .



La función de macro de PCA devuelve  $U$  en las primeras  $n$  columnas y los elementos diagonales de  $\Sigma$  en la última columna.

## Ejemplos

TEMP = PCA(5)Crea dos nuevas columnas denominadas TEMP y VX, que contienen los valores -1 y 0, respectivamente.
TEMP = PCA(V1)  Crea dos nuevas columnas denominadas TEMP y VX. La columna TEMP contiene el valor uno y la columna VX contiene los correspondientes valores propios.
TEMP = PCA(V1:V3)  Crea cuatro columnas nuevas denominadas TEMP, VX, VY y VZ. Los valores de las tres columnas contienen un autovector por fila para los datos en las columnas V1-V3. El valor en la columna VZ contiene los correspondientes valores propios.

## Funciones relacionadas

Función	Descripción
PCA_FEATURES	Extrae n características del rango de datos especificado

---

## PCA\_FEATURES



### Sintaxis

```
PCA_FEATURES(num_features, data [, PCA(base_data)])
```

### Parámetros

num\_features

El número de características para extraer del rango de datos especificado utilizando el análisis de componente principal (PCA). Este valor debe ser un entero positivo que se encuentre entre uno y el número de columnas del rango de datos especificado por data.

data

Los valores numéricos de los cuales se deben extraer características. Este puede ser una columna, un rango de celdas o una expresión que evalúa a cualquiera de los anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

PCA(base\_data)

Si se proporciona este parámetro opcional, PCA se realiza en este rango de datos de base\_data y los autovectores resultantes se utilizan para extraer características del rango de datos de data. Para obtener la definición de formato de base\_data (al

igual que data), consulte la sección "Macro Function Parameters" en el capítulo de esta guía para el producto de IBM . El número de columnas en base\_data debe ser el mismo que el número de columnas en data.

## Descripción

PCA\_FEATURES extrae las primeras num\_features características del rango de datos especificado. Devuelve num\_features columnas utilizando los autovectores generados por el análisis de componente principal en el rango de datos base\_data, si se proporcionó. Si no se proporcionó, utiliza data para generar los autovectores. En este caso, data se normaliza automáticamente utilizando el método de promedio cero, variante de unidad, antes del análisis de componente principal.

Las características se calculan de la siguiente manera:

- El rango de datos data se normaliza automáticamente utilizando el método de promedio cero, variante de unidad. En otras palabras,  $\text{PCA\_FEATURES}(\text{num\_features}, \text{data})$

es equivalente a

$\text{PCA\_FEATURES}(\text{num\_features}, \text{data}, \text{PCA}(\text{data}, \text{COL}))$

Ninguna normalización de data se proporciona automáticamente. Para normalizar data utilizando NORM\_ZSCORE, puede especificar lo siguiente:

$\text{PCA\_FEATURES}(\text{num\_features}, \text{data}, \text{PCA}(\text{NORM\_ZSCORE}(\text{data}, \text{COL})))$

- El análisis de componente principal se realiza en el rango de datos normalizado para generar sus autovectores (consulte los detalles descritos para la función de macro PCA). Esto se produce automáticamente para data si base\_data no se proporciona. Se realiza mediante la llamada explícita a la función de macro PCA si base\_data se proporciona.
- Cada fila ( $Y_i$ ) del rango de datos (data) se transforma en un nuevo sistema coordinado ( $u_i$ ) que se basa en los primeros autovectores clasificados num\_features ( $m$ ) que componen  $E_m$  :

$$u_i = E_m V_i = \begin{bmatrix} E_{11} & \dots & E_{1n} \\ \dots & \dots & \dots \\ E_{m1} & \dots & E_{mn} \end{bmatrix} \begin{bmatrix} V_1 \\ \dots \\ V_n \end{bmatrix}$$

- Las  $k$  filas de los datos transformados ( $u_1$  a  $u_k$ ) se devuelven ( $n$  columnas).

Si se proporciona el rango de datos base\_data, debe tener el mismo número de columnas que el rango de datos data, de lo contrario, se devuelve un error.

**Nota:** Dado que el cálculo de PCA en un rango de datos puede ser calculado en forma intensiva, la utilización de la función de macro BUFFER en el cálculo de PCA es mucho más eficaz. Por ejemplo: PCA\_FEATURES(num\_features, range, BUFFER(PCA(base\_data)))

## Ejemplos

<p>TEMP = PCA_FEATUES(5, V1:V7)Crea cinco columnas nuevas denominadas TEMP, VW, VX, VY y VZ, que contienen las cinco primeras características del rango de datos V1:V7. El rango de datos V1:V7 se utiliza como base para la transformación.</p>
<p>TEMP = PCA_FEATURES(3, V1:V4, PCA(V10:V13))</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, que contienen las tres primeras características del rango de datos V1:V4. El rango de datos V10:V13 se utiliza como base para la transformación.</p>
<p>TEMP = PCA_FEATURES(3, V1:V4, BUFFER(PCA(V10:V13)))</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, que contienen las tres primeras características del rango de datos V1:V4. El rango de datos V10:V13 se utiliza como base para la transformación. Una vez que los principales componentes del rango de datos V10:V13 se calculan, esos valores se almacenan como constantes. Si los valores de los datos de las columnas V10-V13 cambian, no afectarán a esta definición de función.</p>

## Funciones relacionadas

Función	Descripción
PCA	Calcula los vectores para los componentes principales del rango de datos especificado

---

## POSITION

### Sintaxis

POSITION(colName, pattern [, start [, occurrence]])

### Parámetros

colName

El valor de una columna (debe ser de tipo string).

pattern

El patrón, o cadena, que está buscando.

start

El byte con el que desea iniciar la búsqueda.

occurrence

Especifique un valor para n, donde esté buscando la n (enésima) aparición del patrón a devolver.

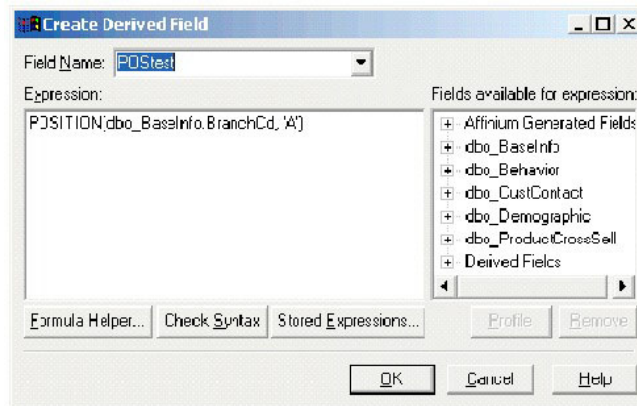
## Descripción

POSITION devuelve la posición de byte inicial de un patrón, o cadena, dentro del valor de una columna (colName) que debe ser de tipo cadena. Si se especifica inicio, empieza a buscar desde allí. La aparición es la enésima aparición de patrón a devolver.

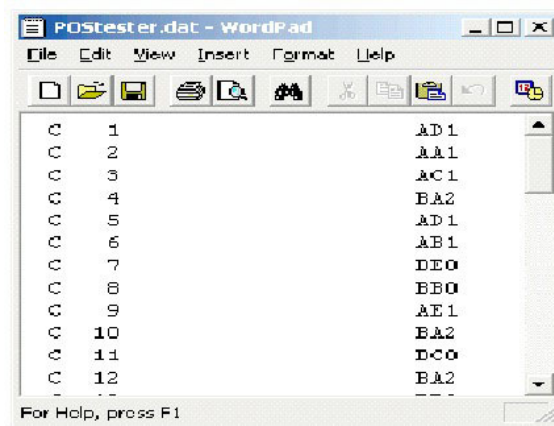
**Nota:** La búsqueda no es sensible a mayúsculas y minúsculas.

## Ejemplos

En el ejemplo siguiente, estamos buscando el patrón o cadena, 'A ', dentro del valor de la columna, dbo\_BaseInfo.BranchCd y asignando el valor devuelto a una POSTest archivada derivada.



El ejemplo siguiente muestra una pocas filas de la tabla con los valores de dbo\_BaseInfo.BranchCd y POSTest que se muestran en conjunto.



Un ejemplo más complejo:

```
STRING_SEG(POSITION(CellCode, "X", 1, 2) + 1,
```

```
STRING_LENGTH(CellCode), CellCode) = "AAA"
```

Esto devuelve filas en las que los valores de CellCode tienen "AAA" al final después de la segunda aparición de "X".

---

## PLUS

### Sintaxis

data PLUS addend data + addend

### Parámetros

data

El rango de celdas que contiene números para añadir. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

addend

El/los número(s) para añadir a todos los valores de la columna especificada. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en data2 debe ser igual al número de columnas en data1, a menos que data2 sea una constante. Para obtener la definición de formato de addend (igual a data), consulte la sección "Parámetros de la función de macro" en el capítulo de esta guía para su producto IBM .

### Descripción

PLUS añade los valores en los dos rangos de datos especificados. Devuelve una columna nueva por cada columna de entrada, cada una de la cuales tiene su columna correspondiente en data1 sumada con la columna correspondiente de data2 (esto es, la primera columna de data1 se suma a la primera columna de data, la segunda columna con la segunda columna, etc.).

Si data2 es una constante, cada valor en data1 se aumenta por ese valor. Si data2 contiene una o más columnas, los cálculos se realizan fila por fila entre una columna de data1 y una columna de data2. La primera fila de data1 se suma al valor de primera fila de data2, la segunda fila con la segunda fila y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** El operador PLUS se puede abreviar con un signo más (+).

Este macro está disponible en IBM Interact.

### Ejemplos

```
TEMP = 3 PLUS 4 o TEMP = 3 + 4
```

Crea una nueva columna denominada TEMP que contiene el valor siete.

TEMP = V1 + 8
Crea una nueva columna denominada TEMP, donde cada valor es el contenido de la columna V1 más ocho.
TEMP = V1 + V1
Crea una nueva columna denominada TEMP que contiene el contenido de la columna V1 multiplicado por dos.
TEMP = V1 + V2
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 más el correspondiente valor de fila de la columna V2.
TEMP = V1:V3 + V4:V6
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores en V1 más los valores de fila correspondientes de la columna V4. La columna VX suma las columnas V2 y V5. La columna VY suma las columnas V5 y V6.
TEMP = V1[10:20] + V2 o TEMP = V1[10:20] + V2[1:11]
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen la suma de los valores de las filas 10-20 de la columna v1 y los valores de las filas 1-11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
MINUS	Resta un rango de datos especificado de otro
SUM o TOTAL	Calcula la suma de un rango de celdas

---

## POW

### Sintaxis

base POW exponent base ^ exponent

### Parámetros

base

Los valores numéricos a elevar a un número exponencial. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de base (igual a data), consulte la sección "Parámetros de la función de macro" en el capítulo de esta guía para su producto IBM .

exponent

El/los número(s) exponencial(es) a los que se deben elevar los valores en data. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. El número de columnas en exponent debe ser igual al número de columnas en base, a menos que base sea una constante. Para obtener la definición de formato del exponent (al igual que de data), consulte la sección "Macro Function Parameters" en el capítulo de esta guía para el producto de IBM .

## Descripción

POW eleva los valores del primer rango de datos a la potencia especificada en el segundo rango de datos (es decir, calcula  $\text{base}^{\text{exponent}}$ ). Devuelve una columna nueva para cada columna de entrada, cada una de las cuales contiene el resultado de la elevación de base a la exponent potencia (es decir, la primera columna de data1 se eleva a la primera columna de data, la segunda columna con la segunda columna, y así sucesivamente).

Si exponent es una constante, cada valor en base se eleva a ese valor. Si exponent contiene una o más columnas, los cálculos se realizan fila por fila entre una columna de base y una columna de exponent. La primera fila de base se suma al valor de primera fila de exponent, la segunda fila con la segunda fila y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** El operador POW se puede abreviar con un circunflejo (^). Por ejemplo, TEMP = 2^8 es equivalente a TEMP = 2 POW 8.

**Nota:** Si el valor  $x$  es demasiado grande o demasiado pequeño, se devuelve un error por desbordamiento. Esto ocurre si  $\text{base}^{\text{exponent}}$  supera el máximo o mínimo de 32 bits del valor de coma flotante.

## Ejemplos

TEMP = 2 POW 3 o TEMP = 2^3
Crea una nueva columna denominada TEMP que contiene el valor ocho.
TEMP = V1 ^ 0.5
Crea una nueva columna denominada TEMP, en donde cada valor es la raíz cuadrada del contenido de la columna V1 (equivalente a SQRT(V1)).
TEMP = V1 ^ V3
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 elevado al valor de fila correspondiente de la columna V2.
TEMP = V1:V3 ^ V4:V6
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores en V1 elevados a los valores de fila correspondientes de la columna V4. La columna VX contiene el resultado de la columna V2 elevado a los valores correspondientes de la columna V5. La columna VY contiene el resultado de la columna V3 elevado a los valores correspondientes de V6.
TEMP = V1[10:20] POW V2 o TEMP = V1[10:20] POW V2[1:11]
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen el resultado de elevar los valores de las filas 10-20 de la columna V1 a los valores de las filas 1-10 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
EXP	Calcula el número natural (e) elevado al contenido de cada celda del rango de datos especificado

Función	Descripción
LN o LOG	Calcula el logaritmo natural del contenido del rango de datos especificado
LN2	Calcula el logaritmo en base 2 del contenido del rango de datos especificado
LN10	Calcula el logaritmo en base 10 del contenido del rango de datos especificado

---

## RANDOM

### Sintaxis

RANDOM(num [, seed])    RANDOM(num, value1, value2 [, seed])

### Parámetros

num

El número de números aleatorios a generar. Este valor debe ser un entero positivo mayor que cero.

value1

Un límite en los números aleatorios a generar. Este puede ser cualquier valor constante o una expresión que evalúe a una constante. Si no se proporciona este parámetro, el valor predeterminado es cero.

value2

El otro límite en los números aleatorios a generar. Este puede ser cualquier valor constante o una expresión que evalúe a una constante. Si no se proporciona este parámetro, el valor predeterminado es uno.

seed

Un valor de inicio opcional que se utiliza para la generación de números aleatorios. Este debe ser entero.

### Descripción

RANDOM genera una columna de números aleatorios. Devuelve una nueva columna que contiene los números aleatorios de num. Si value1 y value2 se han especificado, los números aleatorios se generarán entre esos límites (incluidos). Si no se han especificado, el valor predeterminado es generar valores entre cero y uno. Si se proporciona seed, se utilizará como un inicio para el generador de números aleatorios.

**Nota:** Si seed es mayor o igual que  $2^{32}$ , el valor se sustituye por  $2^{32} - 1$ . Los valores de seed que estén por encima de  $2^{24}$  se redondearán (es decir, se pierde precisión). Por lo tanto, varios valores pueden dar como resultado el mismo valor de seed.



## Ejemplos

TEMP = RANDOM() Crea una nueva columna denominada TEMP que contiene números aleatorios de longitud ilimitada.
TEMP = RANDOM(100) Crea una nueva columna denominada TEMP que contiene 100 números aleatorios que se encuentran entre 0,0 y 1,0.
TEMP = RANDOM(100, 5943049) Crea una nueva columna denominada TEMP que contiene 100 números aleatorios generados desde el número de inicio de 5943049.
TEMP = RANDOM(100, 0, 100) Crea una nueva columna denominada TEMP que contiene 100 números aleatorios que se encuentran entre 0 y 100,0.
TEMP = RANDOM(100, 0, 100, 5943049) Crea una nueva columna denominada TEMP que contiene 100 números aleatorios que se encuentran entre -0 y 100 generados desde el número de inicio de 5943049.

## Funciones relacionadas

Función	Descripción
RANDOM_GAUSS	Devuelve el número especificado de valores aleatorios de una distribución gaussiana
SAMPLE_RANDOM	Devuelve la(s) columna(s) de de las n celdas, cada una de las cuales contiene una muestra aleatoria del rango de datos especificado

---

## RANDOM\_GAUSS

### Sintaxis

RANDOM\_GAUSS(num [, seed]) RANDOM\_GAUSS(num, mean, std [, seed])

### Parámetros

num

El número de números aleatorios a generar. Este valor debe ser un entero positivo mayor que cero.

mean

El promedio del gaussiano. Este puede ser cualquier valor constante o una expresión que evalúe a una constante. Si no se proporciona este parámetro, el valor predeterminado es cero.

std

La desviación estándar del gaussiano. Este puede ser cualquier valor constante o una expresión que evalúe a una constante. Si no se proporciona este parámetro, el valor predeterminado es uno.

seed

Un valor de inicio opcional que se utiliza para la generación de números aleatorios. Este debe ser entero. (Si se proporciona un valor que no es un entero, se utiliza el valor base en su lugar.)

## Descripción

RANDOM\_GAUSS genera una columna de números aleatorios basándose en una distribución gaussiana. Devuelve una nueva columna que contiene los números aleatorios de num. Si se especifica mean y std, los números aleatorios se generarán utilizando una distribución gaussiana con la desviación estándar y promedio especificada. Si no se especifica, el valor predeterminado gaussiano tiene un promedio de cero y una desviación estándar de uno. Si se proporciona seed, se utilizará como un inicio para el generador de números aleatorios.

## Ejemplos

TEMP = RANDOM_GAUSS(100)
Crea una nueva columna denominada TEMP que contiene 100 valores muestreados aleatoriamente desde un gaussiano de promedio cero, desviación unidad-estándar.
TEMP = RANDOM_GAUSS(500, 3)
Crea una nueva columna denominada TEMP que contiene 100 valores muestreados aleatoriamente desde un gaussiano de promedio cero, desviación unidad-estándar. El número 3 se utiliza como valor de inicio para el generador de números aleatorios.
TEMP = RANDOM_GAUSS(5000, 100, 32)
Crea una nueva columna denominada TEMP que contiene 5000 valores muestreados aleatoriamente de un gaussiano con un promedio de 100 y una desviación estándar de 32.
TEMP = RANDOM_GAUSS(500, -1, 2, 3)
Crea una nueva columna denominada TEMP que contiene 500 valores muestreados aleatoriamente de un gaussiano con un promedio de -1 y una desviación estándar de 2. El número 3 se utiliza como un valor de inicio para el generador de números aleatorios.

## Funciones relacionadas

Función	Descripción
RANDOM	Devuelve el número especificado de valores aleatorios
SAMPLE_RANDOM	Devuelve la(s) columna(s) de de las n celdas, cada una de las cuales contiene una muestra aleatoria del rango de datos especificado

---

## RANK



## Sintaxis

RANK(data [, nbins])

## Parámetros

data

Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa cualquiera de éstos.

nbins

El número de agrupaciones en se dividirá data. El valor predeterminado es diez.

## Descripción

RANK divide los datos en nbins (el valor predeterminado es 10) grupos, cada uno con aproximadamente un número igual de valores distintos y devuelve el grupo en el que cae cada fila de datos. La salida será entre 1 y nbins; si el número de valores distintos de datos es menor que nbins, la salida será entre 1 y el número de valores distintos de datos.

RANK silenciosamente impone un límite superior de 1024\*1024 en el valor de nbins. Ese mismo número también se utiliza como número máximo de valores distintos para realizar un seguimiento; valores distintos posteriores se incluirán en la mayor agrupación.

## Ejemplos

TEMP=RANK(V6)

Crea una nueva columna denominada TEMP con cada valor, de uno a diez, que representan la agrupación de datos en que cae la fila. En este caso, se aplica el valor predeterminado de cantidad de agrupaciones, 10.

TEMP=RANK(V6,15)

Crea una nueva columna denominada TEMP con cada valor, de uno a quince, que representan la agrupación de datos en que cae la fila.

TEMP = REPEAT(3, V1:V3)

Crea tres nuevas columnas denominadas TEMP, VX y VY. TEMP contiene tres copias de la columna V1, VX contiene tres copias de la columna V2 y VY contiene tres copias de la columna V3. Todas las columnas de longitud desigual se rellenan hasta la longitud de la columna más larga de V1 - V3.

TEMP = REPEAT(10, V1[10:50]:V2)

Crea dos nuevas columnas denominadas TEMP y VX. Los valores en la columna TEMP son 10 copias de las filas 10 a 50 de la columna V1 y los valores en la columna VX son 10 copias de las filas 10 a 50 de la columna V2.

TEMP = REPEAT((1,2,3), (10, 20, 30))

Crea una nueva columna denominada TEMP, que contiene los valores de celda 10, 20, 20, 30, 30, 30.

<p>TEMP = REPEAT(V1, V2)</p> <p>Crea una nueva columna denominada TEMP. El valor de la celda V2[1] se repite V1[1] veces, el valor de la celda V2[2] se repite V1[2] veces y así sucesivamente hasta el final de la columna V1.</p>
<p>TEMP = REPEAT(V1, V2:V3)</p> <p>Crea dos nuevas columnas denominadas TEMP y VX. La columna TEMP contiene copias de celdas en V2; la columna VX contiene copias de las celdas en V3. Hay V1[1] copias de V2[1] y V3[1], V1[2] copias de V2[2] y V3[2]. Este proceso continúa hasta el final de la columna V1 o el final de la columna más larga en data, en función de cuál sea más corta. Las columnas más cortas de data se rellenan con ceros.</p>
<p>TEMP = REPEAT(3, V1, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una siendo una copia de la columna V1.</p>
<p>TEMP = REPEAT(2, V1:V2, COL)</p> <p>Crea cuatro columnas nuevas denominadas TEMP, VX, VY y VZ. La columna TEMP es una copia de la columna V1; la columna VX es una copia de la columna V2, la columna VY es una copia de la columna V1; y la columna VZ es una copia de la columna V2.</p>

## REPEAT



### Sintaxis

REPEAT(num\_times, data [, keyword])

### Parámetros

num\_times

El número de veces que se debe repetir el rango de datos especificado. Este puede ser una constante, una columna o una expresión que evalúa a cualquiera de los anteriores. Todos los valores deben ser enteros positivos.

data

Los valores numéricos a repetir. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

keyword

Esta palabra clave opcional determina cómo se replica el rango de datos especificado. Seleccione una de las opciones siguientes:

ROW - Repite las filas de data verticalmente (valor predeterminado)

COL - Repite columnas de data horizontalmente

Para obtener más detalles sobre la utilización de palabras clave en IBM Campaign, consulte “Especificaciones de formato” en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte “Especificaciones de formato” en la página 27.

## Descripción

REPEAT repite el rango de datos especificado un número de veces, verticalmente (COL) u horizontalmente (ROW).

Con la palabra clave COL, se devuelve una columna nueva para cada columna de entrada que contiene copias de num\_times de data, concatenadas verticalmente. Si las columnas de entrada de data tienen longitudes diferentes, las columnas más cortas se rellenan automáticamente. Las columnas numéricas se rellenan con cero y las columnas de cadena de texto se rellenan con series vacías ("").

Con la palabra clave ROW, el número de columnas devuelto depende de si num\_times es una constante o una columna. Si num\_times es una constante, devuelve num\_times veces el número de columnas en data. Si num\_times es una columna, devuelve el número especificado de copias de cada columna de data, donde el primer valor de fila de num\_times es el número de veces a repetir la primera columna de data; el segundo valor de fila es el número de veces a repetir la segunda columna de data, etc. Cualquier fila que sea mayor que el número de columnas en data se ignora.

## Ejemplos

<pre>TEMP = REPEAT(10, 1) o TEMP = REPEAT(10, 1, ROW)</pre>
<p>Crea una nueva columna denominada TEMP que contiene diez 1.</p>
<pre>TEMP = REPEAT(2, V1)</pre>
<p>Crea una nueva columna denominada TEMP que contiene dos copias de la columna V1 concatenadas verticalmente.</p>
<pre>TEMP = REPEAT(3, V1:V3)</pre>
<p>Crea tres nuevas columnas denominadas TEMP, VX y VY. TEMP contiene tres copias de la columna V1, VX contiene tres copias de la columna V2 y VY contiene tres copias de la columna V3. Todas las columnas de longitud desigual se rellenan hasta la longitud de la columna más larga de V1 - V3.</p>
<pre>TEMP = REPEAT(10, V1[10:50]:V2)</pre>
<p>Crea dos nuevas columnas denominadas TEMP y VX. Los valores de la columna TEMP son 10 copias de las filas 10-50 de la columna V1 y los valores de la columna VX son 10 copias de las filas 10-50 de la columna V2.</p>
<pre>TEMP = REPEAT((1,2,3), (10, 20, 30))</pre>
<p>Crea una nueva columna denominada TEMP, que contiene los valores de celda 10, 20, 20, 30, 30, 30.</p>
<pre>TEMP = REPEAT(V1, V2)</pre>
<p>Crea una nueva columna denominada TEMP. El valor de la celda V2[1] se repite V1[1] veces, el valor de la celda V2[2] se repite V1[2] veces y así sucesivamente hasta el final de la columna V1.</p>

```
TEMP = REPEAT(V1, V2:V3)
```

Crea dos nuevas columnas denominadas TEMP y VX. La columna TEMP contiene copias de celdas en V2; la columna VX contiene copias de las celdas en V3. Hay V1[1] copias de V2[1] y V3[1], V1[2] copias de V2[2] y V3[2]. Este proceso continúa hasta el final de la columna V1 o el final de la columna más larga en data, en función de cuál sea más corta. Las columnas más cortas de data se rellenan con ceros.

```
TEMP = REPEAT(3, V1, COL)
```

Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una siendo una copia de la columna V1.

```
TEMP = REPEAT(2, V1:V2, COL)
```

Crea cuatro columnas nuevas denominadas TEMP, VX, VY y VZ. La columna TEMP es una copia de la columna V1; la columna VX es una copia de la columna V2, la columna VY es una copia de la columna V1; y la columna VZ es una copia de la columna V2.

---

## ROTATE\_LEFT



### Sintaxis

```
ROTATE_LEFT(num_cols, data)
```

### Parámetros

num\_cols

El número de columnas a girar hacia la izquierda. Este valor debe ser un entero no negativo. El valor de cero copia las filas sin rotación.

data

Los valores numéricos a girar hacia la izquierda. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

ROTATE\_LEFT gira el rango de datos especificado hacia la izquierda un número de columnas. Devuelve una columna nueva para cada columna de entrada, cada una conteniendo una copia de la columna de entrada correspondiente girada hacia la izquierda num\_cols posiciones. Las columnas que se han girado desde la izquierda están encapsuladas en la derecha.

**Nota:** ROTATE\_LEFT sólo funciona con datos numéricos. Ninguno de los datos proporcionados en el parámetro data pueden ser texto ASCII.

## Ejemplos

```
TEMP = ROTATE_LEFT(1, MERGE(1, 2, 3))
```

Crea tres nuevas columnas denominadas TEMP, VX y VY. TEMP contiene el valor dos, VX contiene el valor tres y VY contiene el valor uno.

```
TEMP = ROTATE_LEFT(0, V1:V3)
```

Crea tres nuevas columnas denominadas TEMP, VX y VY. TEMP es una copia de la columna V1, VX es una copia de la columna V2 y VY es una copia de la columna V3.

```
TEMP = ROTATE_LEFT(4, V1:V3)
```

Crea tres nuevas columnas denominadas TEMP, VX y VY. TEMP es una copia de la columna V2, VX es una copia de la columna V3 y VY es una copia de la columna V1.

```
TEMP = ROTATE_LEFT(1, V1[10:50]:V2)
```

Crea dos nuevas columnas denominadas TEMP y VX, cada una con valores en las primeras 41 filas (las otras celdas están vacías). Los valores de la columna TEMP son los valores de las filas 10-50 de la columna V2 y los valores de la columna VX son los valores de las filas 10-50 de la columna V1.

## Funciones relacionadas

Función	Descripción
ROTATE_RIGHT	Gira las columnas del rango de datos especificado hacia la derecha

---

## ROTATE\_RIGHT



### Sintaxis

```
ROTATE_RIGHT(num_cols, data)
```

### Parámetros

num\_cols

El número de columnas a girar hacia la derecha. Este valor debe ser un entero no negativo. El valor de cero copia las filas sin rotación.

data

Los valores numéricos a girar hacia la derecha. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

ROTATE\_RIGHT gira el rango de datos especificado hacia la derecha un número de columnas. Devuelve una columna nueva para cada columna de entrada, cada una

conteniendo una copia de la columna de entrada correspondiente girada hacia la derecha `num_cols` posiciones. Las columnas que se han girado desde la derecha están encapsuladas en la izquierda.

**Nota:** `ROTATE_RIGHT` sólo funciona con datos numéricos. Ninguno de los datos proporcionados en el parámetro `data` pueden ser texto ASCII.

## Ejemplos

<pre>TEMP = ROTATE_RIGHT(1, MERGE(1, 2, 3))</pre> <p>Crea tres nuevas columnas denominadas <code>TEMP</code>, <code>VX</code> y <code>VY</code>. <code>TEMP</code> contiene el valor tres, <code>VX</code> contiene el valor uno y <code>VY</code> contiene el valor dos.</p>
<pre>TEMP = ROTATE_RIGHT(0, V1:V3)</pre> <p>Crea tres nuevas columnas denominadas <code>TEMP</code>, <code>VX</code> y <code>VY</code>. <code>TEMP</code> es una copia de la columna <code>V1</code>, <code>VX</code> es una copia de la columna <code>V2</code> y <code>VY</code> es una copia de la columna <code>V3</code>.</p>
<pre>TEMP = ROTATE_RIGHT(4, V1:V3)</pre> <p>Crea tres nuevas columnas denominadas <code>TEMP</code>, <code>VX</code> y <code>VY</code>. <code>TEMP</code> es una copia de la columna <code>V3</code>, <code>VX</code> es una copia de la columna <code>V1</code> y <code>VY</code> es una copia de la columna <code>V2</code>.</p>
<pre>TEMP = ROTATE_RIGHT(1, V1[10:50]:V2)</pre> <p>Crea dos nuevas columnas denominadas <code>TEMP</code> y <code>VX</code>, cada una con valores en las primeras 41 filas (las otras celdas están vacías). Los valores de la columna <code>TEMP</code> son los valores de las filas 10-50 de la columna <code>V2</code> y los valores de la columna <code>VX</code> son los valores de las filas 10-50 de la columna <code>V1</code>.</p>

## Funciones relacionadas

Función	Descripción
<code>ROTATE_LEFT</code>	Gira las columnas del rango de datos especificado hacia la izquierda

---

## ROUND

### Sintaxis

`ROUND(data)`

### Parámetros

`data`

Los valores numéricos a redondear. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de `data`, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

`ROUND` redondea los valores en el rango de datos especificado al entero más cercano. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene el valor redondeado de números en la columna de entrada



correspondiente. Los números que se encuentran exactamente en la mitad, se redondean hacia arriba (por ejemplo, 2,5 se redondea a 3,0 y -2,5 se redondea a -2,0).

## Ejemplos

TEMP = ROUND(3.2)
Crea una nueva columna denominada TEMP que contiene el valor tres.
TEMP = ROUND(V1)
Crea una nueva columna denominada TEMP, donde cada valor es el valor redondeado del contenido de la columna V1.
TEMP = ROUND(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los valores redondeados del contenido de la columna V1, los valores de la columna VX son los valores redondeados del contenido de la columna V2 y los valores de la columna VY son los valores redondeados del contenido de la columna V3.
TEMP = ROUND(V1[10:20])
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los valores redondeados en las filas 10-20 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = ROUND(V1[1:5]:V2)
Crea dos nuevas columnas denominadas TEMP y VX, con valores en las filas 1-5 (las otras celdas están vacías). Los valores en la columna TEMP son los valores redondeados de las filas correspondientes de la columna V1, y los valores en la columna VX son los valores redondeados de las filas correspondientes de la columna V2.

## Funciones relacionadas

Función	Descripción
INT	Calcula el valor entero (redondeado a la baja) del contenido del rango de datos especificado
MOD	Calcula el módulo del contenido del rango de datos especificado
TRUNCATE	Devuelve la parte no fraccionaria de cada valor del rango de datos especificado

---

## ROWNUM

### Sintaxis

ROWNUM()

### Descripción

ROWNUM genera números secuenciales desde uno hasta el número de registros. El número del primer registro es uno, dos para el segundo registro, etc.

**Nota:** El número máximo de registros que puede manejar ROWNUM es de dos mil millones.

---

## RTRIM

### Sintaxis

RTRIM(data)

### Parámetros

data

### Descripción

RTRIM elimina los caracteres de espacio final de cada valor de cadena en el rango de datos especificado, devolviendo la cadena convertida. Devuelve una nueva columna para cada columna de entrada.

Este macro está disponible en IBM Interact.

### Ejemplos

```
Temp = RTRIM "gold "
```

Crea una nueva cadena denominada Temp que contiene "gold".

---

## SAMPLE\_RANDOM



### Sintaxis

SAMPLE\_RANDOM(num\_samples, data [, seed])

### Parámetros

num\_samples

El número de muestras a tomar de cada columna en el rango de datos especificado.

data

Los valores para realizar el muestreo de forma aleatoria. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

seed

Un valor de inicio opcional que se utiliza para la generación de números aleatorios. Este debe ser entero. (Si se proporciona un valor que no es un entero, se utiliza el valor base en su lugar.)

## Descripción

SAMPLE\_RANDOM realiza un muestreo aleatorio del rango de datos especificado. Devuelve una columna nueva para cada columna de entrada, cada una de las cuales contiene num\_samples números muestreados aleatoriamente desde la columna de entrada correspondiente de data. Las muestras se toman en el orden en que aparecen en cada columna (es decir, los valores de datos permanecerán en el mismo orden relativo entre sí). Si se proporciona seed, se utilizará como un inicio para el generador de números aleatorios.

## Ejemplos

TEMP = SAMPLE_RANDOM(100, 3)
Crea una nueva columna denominada TEMP con 100 celdas que contienen el valor 3.
TEMP = SAMPLE_RANDOM(100, V1)
Crea una nueva columna denominada TEMP que contiene 100 valores, donde cada valor es una muestra aleatoria del contenido de la columna V1.
TEMP = SAMPLE_RANDOM(50, V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene 50 valores. Los valores de la columna TEMP son muestras aleatorias de la columna V1, los valores de la columna VX son muestras aleatorias de la columna V2 y los valores de la columna VY son muestras aleatorias de la columna V3.
TEMP = SAMPLE_RANDOM(100, V1[10:50]:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene 41 valores. Los valores de la columna TEMP son muestras aleatorias de las celdas 10-50 de la columna V1, los valores de la columna VX son muestras aleatorias de las celdas 10-50 de la columna V2 y los valores de la columna VY son muestras aleatorias de las celdas 10-50 de la columna V3. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
RANDOM	Devuelve el número especificado de valores aleatorios
RANDOM_GAUSS	Devuelve el número especificado de valores aleatorios de una distribución gaussiana
SUBSAMPLE	Reduce los datos al devolver cada enésimo valor de fila

---

## SELECT



### Sintaxis

SELECT(col\_nums, data) SELECT(from\_col, data) SELECT(from\_col, to\_col, data)

## Parámetros

`col_nums`

El nombre de una columna que contiene los números de columna para extraer del rango de datos especificado (por ejemplo, una columna que contiene los números 1, 3, 4 y 7, extrae la primera, tercera, cuarta y séptima columna del rango de datos especificado). Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de `col_nums` (al igual que `data`), consulte la sección "Macro Function Parameters" en el capítulo de esta guía para el producto de IBM .

`from_col`

La posición numérica de la columna para extraer del rango de datos especificado. Un valor de uno extrae la primera columna de un rango de datos especificado.

`to_col`

Si este parámetro está proporcionado, `from_col` se utiliza como un punto de partida y debe ser una columna o un rango de celdas. El punto final es especificado por `to_col`. Este valor debe ser mayor que `from_col`.

`data`

El rango de celdas que contiene la columna(s) a extraer. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de `data`, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

SELECT devuelve las columnas especificadas de un rango de datos. Las columnas para seleccionar pueden especificarse en diferentes formas. El parámetro `col_nums` contiene números de columna para extraer del rango de datos especificado. para extraer una sola columna, utilice `from_col`. Para extraer un rango de columnas consecutivas, especifique `from_col` como un punto de partida (debe ser una columna o un rango de celdas) y `to_col` como una columna final.

Si `data` es escalar (es decir, una constante o una variable que contiene un solo valor), la selección de la primera columna devuelve una columna nueva que contiene esa constante. La selección de otra columna desde una constante devuelve una columna que contiene ???.

Esta macro es a menudo incluida en funciones más complejas.

**Nota:** Para extraer varias columnas que no sean adyacentes, utilice la función de macro COLUMN para crear una columna que contenga los números de columna para seleccionar. Vea el ejemplo siguiente.

## Ejemplos

```
TEMP = SELECT(1, 3)Crea una nueva columna denominada TEMP que contiene el número tres.
```

TEMP = SELECT(1, V1) o TEMP = SELECT(1, V1:V3)
Crea una nueva columna denominada TEMP que es una copia de la columna V1.
TEMP = SELECT(2, 4, V1:V5)
Crea tres nuevas columnas denominadas TEMP, VX y VY. TEMP es una copia de la columna V2, VX es una copia de la columna V3 y VY es una copia de la columna V4.
TEMP = SELECT(COLUMN(1,4), V6:V10)
Crea dos nuevas columnas denominadas TEMP y VX. TEMP es una copia de la columna V6 y VX es una copia de la columna V9.
TEMP = SELECT(COLUMN(1,4), V6[25:74]:V10)
Crea dos nuevas columnas denominadas TEMP y VX, cada una de las cuales contiene 50 valores. TEMP es una copia de las celdas 25-74 de la columna V6 y VX es una copia de las celdas 25-74 de la columna V9. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
EXTRACT	Extrae filas de acuerdo a los valores de una columna de predicado
MERGE	Crea un grupo de datos concatenando horizontalmente los valores de entrada

---

## SIGN

### Sintaxis

SIGN(data)

Parámetros

data

Los valores numéricos para calcular el signo de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

SIGN prueba el signo de los valores en el rango de datos especificado. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene el signo de números en la columna de entrada correspondiente. Se devuelve un uno positivo para todos los valores mayores que cero; un uno negativo se devuelve para todos los valores menores que cero; se devuelve cero para valores de cero.

### Ejemplos

TEMP = SIGN(-3)
Crea una nueva columna denominada TEMP que contiene el valor -1.

TEMP = SIGN(MERGE(3, 2, 0))
Crea tres nuevas columnas denominadas TEMP, VX y VY. TEMP contiene el valor 1, VX contiene el valor -1 y VY contiene el valor 0.
TEMP = SIGN(V1)
Crea una nueva columna denominada TEMP, donde cada valor es el signo de los contenidos de la columna V1.
TEMP = SIGN(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los signos de los contenidos de la columna V1, los valores de la columna VX son los signos de los contenidos en la columna V2 y los valores de la columna VY son los signos de los contenidos de la columna V3.
TEMP = SIGN(V1[10:20])
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los signos de los valores de las filas 10 a 20 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = SIGN(V1[10:50]:V2)
Crea dos nuevas columnas denominadas TEMP y VX, cada una con valores en las filas 1 a 41 (las otras celdas están vacías). Los valores en la columna TEMP son los signos de los valores en las fila 10-50 de la columna V1 y los valores en la columna VX son los signos de los valores en las filas 10-50 de la columna V2.

## Funciones relacionadas

Función	Descripción
ABS	Calcula el número NOT lógico del contenido del rango de datos especificado valor absoluto del contenido del rango de datos especificado
INVERSE	Calcula el negativo del contenido del rango de datos especificado

---

## SIN

### Sintaxis

SIN(data [, units\_keyword])

### Parámetros

data

Los valores numéricos para calcular el seno de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

units\_keyword

Esta palabra clave opcional determina si los valores de entrada y los resultados se interpretan como grados o radianes. Seleccione una de las opciones siguientes:

RADIAN - Realiza los cálculos en radianes (valor predeterminado)

DEGREE - Realiza los cálculos en grados

Si no se especifica este parámetro, el valor predeterminado es radianes. (Para convertir de radianes a grados, divida por PI y multiplíquelo por 180.)

Para obtener más detalles sobre la utilización de las palabras claves en IBM Campaign, consulte “Especificaciones de formato” en la página 9.

Para obtener más detalles sobre la utilización de palabrasclave en IBM PredictiveInsight, consulte “Especificaciones de formato” en la página 27.

## Descripción

SIN calcula el seno de los valores del rango de datos especificado. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene el seno de números en la columna de entrada correspondiente.

## Ejemplos

<p>TEMP = SIN(PI/2) o TEMP = SIN(PI/2, 0) o TEMP = SIGN(PI/2, RADIAN)</p> <p>Crea una nueva columna denominada TEMP que contiene el valor uno.</p>
<p>TEMP = SIN(V1)</p> <p>Crea una nueva columna denominada TEMP, donde cada valore es el seno (expresado en radianes) del contenido de la columna V1.</p>
<p>TEMP = SIN(V1:V3, 1) o TEMP = SIN(V1:V3, DEGREE)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los senos de los contenidos de la columna V1, los valores de la columna VX son los senos de los contenidos en la columna V2 y los valores de la columna VY son los senos de los contenidos de la columna V3. Todos los valores se expresan en grados.</p>
<p>TEMP = SIN(V1[10:50]:V2)</p> <p>Crea dos nuevas columnas denominadas TEMP y VX, cada una con valores en las filas 1 a 41 (las otras celdas están vacías). Los valores en la columna TEMP son los senos de los valores en las fila 10-50 de la columnaV1 y los valores en la columna VX son los senos de los valores en las filas 10-50 de la columna V2. Todos los valores se expresan en radianes.</p>

## Funciones relacionadas

Función	Descripción
ASIN	Calcula el número NOT lógico del contenido del rango de datos especificado arcoseno del contenido del rango de datos especificado
COS	Calcula el coseno del contenido del rango de datos especificado
SINH	Calcula el seno hiperbólico del contenido del rango de datos especificado

Función	Descripción
TAN	Calcula la tangente del contenido del rango de datos especificado

## SINH

### Sintaxis

SINH(data [, units\_keyword])

### Parámetros

data

Los valores numéricos para calcular el seno hiperbólico de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

units\_keyword

Esta palabra clave opcional determina si los valores de entrada y los resultados se interpretan como grados o radianes. Seleccione una de las opciones siguientes:

RADIAN - Realiza los cálculos en radianes (valor predeterminado)

DEGREE - Realiza los cálculos en grados

Si no se especifica este parámetro, el valor predeterminado es radianes. (Para convertir de radianes a grados, divida por PI y multiplíquelo por 180.)

Para obtener más detalles sobre la utilización de las palabras claves en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

### Descripción

SINH calcula el seno hiperbólico de los valores en el rango de datos especificados. Devuelve una nueva columna para cada columna de entrada, cada una contiene un seno hiperbólico de números en la correspondiente columna de entrada. Para  $x$  en radianes, el seno hiperbólico de un número es:

$$\sinh(x) = \frac{e^x - e^{-x}}{2}$$

Donde  $e$  es el número natural, 2.7182818.



**Nota:** Si el valor  $x$  es demasiado grande, se devuelve un error de desbordamiento. Esto sucede si  $\sinh(x)$  excede el valor máximo de coma flotante de 32-bit.

## Ejemplos

<p>TEMP = SINH(1) o TEMP = SINH(1, 0) o TEMP = SINH(1, RADIAN)</p> <p>Crea una nueva columna denominada TEMP que contiene el valor 1.18.</p>
<p>TEMP = SINH(V1)</p> <p>Crea una nueva columna denominada TEMP, donde cada valor es un seno hiperbólico (en radianes) de los contenidos de la columna V1.</p>
<p>TEMP = SINH(V1:V3, 1) o TEMP = SINH(V1:V3, GRADO)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los senos hiperbólicos de los contenidos de la columna V1, los valores de la columna VX son los senos hiperbólicos de los contenidos de la columna V2 y los valores de la columna VY son los senos hiperbólicos de los contenidos de la columna V3. Todos los valores se expresan en grados.</p>
<p>TEMP = SINH(V1[10:50]:V2)</p> <p>Crea dos nuevas columnas denominadas TEMP y VX, cada una con valores en las filas 1 a 41 (las otras celdas están vacías). Los valores en la columna TEMP son los senos hiperbólicos de los valores en las fila 10-50 de la columna V1 y los valores en la columna VX son los senos hiperbólicos de los valores en las filas 10-50 de la columna V2. Todos los valores se expresan en radianes.</p>

## Funciones relacionadas

Función	Descripción
COSH	Calcula el número NOT lógico del contenido del rango de datos especificado coseno hiperbólico del contenido del rango de datos especificado
SIN	Calcula el número NOT lógico del contenido del rango de datos especificado seno del contenido del rango de datos especificado
TANH	Calcula la tangente hiperbólica del contenido del rango de datos especificado

## DESVÍO



### Sintaxis

SKEW(data [, keyword])

### Parámetros

data

Los valores numéricos para calcular el desvío de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte el

apartado "Parámetros de función de macro" del capítulo en esta guía para el producto IBM . Tiene que haber al menos tres valores en data.

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las opciones siguientes:

ALL - Realiza el cálculo en todas las celdas de data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de data

Para obtener más detalles sobre la utilización de las palabras claves en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

## Descripción

SKEW calcula el desvío de la distribución de todas las celdas en el rango de datos especificado. El desvío mide el grado de asimetría de una distribución de su media. Un valor de desvío positivo indica una distribución con una cola asimétrica que se inclina hacia valores más positivos; un desvío negativo indica una distribución con una cola asimétrica que se inclina hacia valores más negativos. Un valor de desvío cero indica que la distribución es simétrica sobre su media.

El desvío se calcula de la siguiente manera:

$$\frac{n}{(n-1)(n-2)} \sum_1^n \left( \frac{x_i - \text{mean}}{\sigma} \right)^3$$

Donde *n* es el número de muestras en la distribución, *el promedio* es el promedio y  $\sigma$  es la desviación estándar de la distribución. Para calcular el desvío se deben proporcionar un mínimo de tres valores de datos.

**Nota:** Si la desviación estándar  $\sigma = 0$ , DESVÍO devuelve cero.

## Ejemplos

```
TEMP = DESVÍO(3) o TEMP = DESVÍO(3, TODO)
```

Crea una nueva columna denominada TEMP que contiene el valor cero.

```
TEMP = SKEW(MERGE(3, 7, 2))
```

Crea una nueva columna denominada TEMP que contiene el valor 0.14.

TEMP = SKEW(V1)
Crea una nueva columna denominada TEMP que contiene un valor único el cual es el desvío de los contenidos de la columna V1.
TEMP = DESVÍO(V1:V3)
Crea una nueva columna denominadaTEMP que contiene un valor único el cual es el desvío de los contenidos de las columnasV1, V2 y V3.
TEMP = DESVÍO(V1[10:20])
Crea una nueva columna denominada TEMP que contiene un valor único el cual es el desvío de las celdas en las filas 10-20 de la columna V1.
TEMP = SKEW(V1[1:5]:V4)
Crea una nueva columna denominada TEMP que contiene un valor único el cual es el desvío de las celdas en las filas 1-5 de las columnas V1 por V4.
TEMP = SKEW(V1:V3, COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY. El valor único en la columna TEMP es el desvío de los contenidos de la columna V1, el valor único en la columna VX es el desvío de los contenidos de la columna V2 y el valor único en la columna VY es el desvío de los contenidos de la columna V3.
TEMP = SKEW(V1[1:5]:V3,COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene un valor único. El valor en la columna TEMP es el desvío de las celdas en las filas 1-5 de la columna V1, el valor en la columna VX es el desvío de las celdas en las filas 1-5 de la columna V2 y el valor en la columna VY es el desvío de las celdas en las filas 1-5 de la columnaV3.
TEMP = SKEW(V1:V3, ROW)
Crea una nueva columna denominada TEMP donde cada entrada de celda es el desvío de la fila correspondiente en las columnas V1, V2 y V3.
TEMP = SKEW(V1[10:50]:V3, ROW)
Crea una nueva columna denominada TEMP, donde las primeras 41 celdas contienen el desvío de los valores en las filas 10-50 de las columnas V1 hasta V3. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
AVG o MEAN	Calcula la media aritmética o el promedio de un rango de celdas
KURTOSIS	Calcula la curtosis de un rango de celdas
STAT	Calcula desde el primer al cuarto momento del rango de datos especificado
VARIANCE	Calcula la varianza de un rango de celdas

SLIDE\_WINDOW



## Sintaxis

SLIDE\_WINDOW(width, data [, increment])

## Parámetros

width

El tamaño (número vertical de filas) de la ventana deslizante.

data

El rango de celdas a utilizar para eludir una ventana para generar datos. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

increment

El número de filas para incrementar cada vez; el valor predeterminado es uno. Este debe ser un entero mayor que cero (y menor que la longitud de datos).

## Descripción

SLIDE\_WINDOW utiliza un tamaño de ventana especificado y elude el rango de datos especificado al crear patrones desde los valores de datos de la ventana. Devuelve (el número de columnas x de entrada anchura) columnas. La venta deslizante comienza en la parte superior de los data y cubre width filas. Los valores de datos en esta ventana (lectura de izquierda a derecha, de arriba hacia abajo) se concatenan entre sí para formar una fila única en el rango de datos de salida. La ventana deslizante baja por increment filas cada vez.

Por ejemplo, suponga que las columnas V1 y V2 contienen los siguientes datos:

```
10 2 20 3 30 4 40 5 50 ...
```

Entonces la expresión V3=SLIDE\_WINDOW(2, V1:V2) produce la siguiente salida en las columnas V3:V6:

```
10 2 20 2 20 3 30 3 30 4 40 ...
```

Las primeras dos filas crean la primera fila de salida. A continuación, la ventana se desliza hacia abajo para crear el siguiente patrón, etc.

**Nota:** Esta función es útil para la creación de patrones a partir de datos de las series de tiempo.

## Ejemplos

```
TEMP = SLIDE_WINDOW(1, V1)
```

Crea una nueva columna denominada TEMP que contiene una copia de valores en la columna V1.

TEMP = SLIDE_WINDOW(3, V1:V3)
Crea nueve columnas nuevas con cada fila que contiene una ventana de datos de tres por tres desde las columnas V1:V3. Las filas 1-3 de la entrada de la primera fila de salida, las filas 2-4 forman la segunda, etc.
TEMP = SLIDE_WINDOW(2, V1:V3[10:20])
Crea seis columnas nuevas con cada fila que contiene una ventana de datos de tres por tres de las filas 10-20 de las columnas V1:V3. Las filas 10-11 de la entrada forman la primera fila de salida, las filas 11-12 de la segunda, etc.
TEMP = SLIDE_WINDOW(2, FUSIONA(V1, V3, V5))
Crea seis nuevas columnas con cada fila que contiene una ventana de datos de tres por tres de las columnas V1, V3 y V5. Las filas 1-2 de la entrada forman la primera fila de la salida, las filas 2-3 la segunda, etc.
TEMP = SLIDE_WINDOW(1, V1:V3, 2)
Crea tres nuevas columnas, donde la primera fila contiene datos de V1[1]:V3, la segunda fila contiene datos de V1[3]:V3, la tercera fila contiene datos de V1[5]:V3 y así sucesivamente ( se omiten filas alternadas).
TEMP = SLIDE_WINDOW(10, V1, 10)
Crea diez nuevas columnas, donde la primera fila contiene datos de V1[1:10], la segunda fila contiene datos de V1[11:20], la tercera fila contiene datos de V1[21:30] y así sucesivamente.
TEMP = SLIDE_WINDOW(3, V1:V2, 5)
Crea seis nuevas columnas, donde cada fila contiene una ventana de datos de dos por tres de las columnas V1:V2. La primera fila contiene datos de V1[1:3]:V2; la segunda fila contiene datos de V1[6:8]:V2, la tercera fila contiene datos de V1[11:13]:V2 y así sucesivamente.

## Funciones relacionadas

Función	Descripción
GRID	Devuelve una cuadrícula de todas las combinaciones de valores posibles (una por fila)

---

## SORT



### Sintaxis

SORT(column [, keyword]) SORT(column, data [, keyword])

### Parámetros

column

En este primer formato (no se proporcionan datos), esta es la columna de datos para clasificar (numérica o de texto). Esta puede ser una constante, una columna, un rango de celdas de una única columna o una expresión que evalúa algunas de las anteriores. Estos rangos de datos no pueden contener más de  $2^{29}$  valores.

data

Cuando se proporciona este parámetro, son los datos para clasificar con la utilización de `column` como el criterio de clasificación (`data` pueden contener columnas de datos numéricos y de texto). El parámetro de datos puede ser una columna, un rango de celdas o una expresión que evalúa alguno de los elementos anteriores. Para obtener la definición de formato de `data`, consulte el apartado "Parámetros de función de macro" del capítulo en esta guía para el producto IBM . Estos rangos de datos no pueden contener más de 2<sup>29</sup> filas.

keyword

Esta palabra clave opcional determina si se clasifican los valores en orden de aumento (mínimo a máximo) o de disminución (máximo a mínimo). Seleccione una de las siguientes opciones:

ASCEND - Clasificación de datos en orden ascendente (aumento) (valor predeterminado)

DESCEND - Clasificación de datos en orden descendente (disminución)

Para obtener más detalles sobre la utilización de palabras clave en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

## Descripción

`SORT` clasifica los valores en el rango de datos especificado (ya sea `column` o `data` basado en `column`) basado en el valor de `keyword` (ASCEND o DESCEND). Devuelve una columna nueva para cada columna de entrada a ordenar. Si se proporciona solamente `column`, los valores en `column` se ordenarán en orden ascendente o descendente como especifica el parámetro de palabra clave. Para una columna de texto, el orden ascendente es el orden alfabético (a-z) y el descendente es el orden inverso (z-a). Si también se proporcionan los datos, se clasificarán con la utilización de la `column` como criterio de clasificación.

**Nota:** Si se proporciona un rango de celdas de una única columna para `column`, para ordenar las filas *correspondientes* de datos, debe especificar el mismo rango de celdas para los datos. De lo contrario, el valor predeterminado es ordenar las primeras filas *n* de los datos. Por ejemplo, para ordenar las filas correspondientes, especificar: `TEMP = SORT(V1[100:200], V2[100:200]:V5)` De lo contrario, `TEMP = SORT(V1[100:200], V2:V5)` es equivalente a: `TEMP = SORT(V1[100:200], V2[1:101]:V5)`

## Ejemplos

```
TEMP = SORT(COLUMN(5, 3, 2, 4, 1)) o TEMP = SORT(COLUMN(5, 3, 2, 4, 1), ASCEND)
```

Crea una nueva columna denominada `TEMP` que contiene los valores 1, 2, 3, 4 y 5.

```
TEMP = SORT(COLUMN("b", "c", "a"))
```

Crea una nueva columna denominada `TEMP` que contiene las cadenas a, b y c.

TEMP = SORT(10...15, DESCEND)
Crea una nueva columna denominada TEMP que contiene los valores 15, 14, 13, 12, 11 y 10.
TEMP = SORT(V1)
Crea una nueva columna denominada TEMP que contiene los valores en la columna V1 ordenados en forma ascendente.
TEMP = SORT(V1, V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los contenidos ordenados de la columna V1 en orden ascendente. Los valores de la columna VX son los contenidos correspondientes de la columna V2 y los valores de la columna VY son los contenidos correspondientes de la columna V3.
TEMP = SORT(V1[10:20], DESCEND)
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen los valores ordenados de las celdas en las filas 10-20 de la columna V1 en orden descendente. Las otras celdas de TEMP están vacías.
TEMP = SORT(V1[5:10], V2) o TEMP = SORT(V1[5:10], V2[1:6])
Crea una nueva columna denominada TEMP, donde las primeras 6 celdas contienen los valores de las filas 1-6 de la columna V2, clasificadas en orden descendente de las celdas 5-10 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = SORT(V1[5:10], V2[5:10])
Crea una nueva columna denominada TEMP, donde las primeras 6 celdas contienen los valores ordenados de las celdas en las filas 5-10 de la columna V2 de acuerdo a un orden descendente de las celdas 5-10 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = SORT(V1[10:50], V2:V3) o TEMP = SORT(V1[10:50], V2[1:41]:V3)
Crea dos nuevas columnas denominadas TEMP y VX, cada una con valores en las filas 1 a 41 (las otras celdas están vacías). Los valores en la columna TEMP son los valores de las filas 1 a 41 de la columna V2, ordenados de acuerdo a las filas 10-50 de la columna V1. Del mismo modo, los valores en la columna VX son los valores de las filas 1 a 41 de la columna V3, ordenados de acuerdo a las filas 10-50 de la columna V1. La columna V1 se clasifica en orden ascendente.

---

## SQRT

### Sintaxis

SQRT(data)

Parámetros

data

Los valores numéricos para calcular la raíz cuadrada de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

SQRT calcula la raíz cuadrada de los valores en el rango de datos especificado. Devuelve una nueva columna para cada columna de entrada, cada una contiene la raíz cuadrada positiva de números en la correspondiente columna de entrada.

**Nota:** Su un valor en el rango de datos especificado es negativo, se devuelve un ??? para esa celda.

## Ejemplos

TEMP = SQRT(2)
Crea una nueva columna denominada TEMP que contiene el valor 1.41.
TEMP = SQRT(V1)
Crea una nueva columna denominada TEMP, donde cada valor el la raíz cuadrada de los contenidos de la columna V1.
TEMP = SQRT(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son las raíces cuadradas de los contenidos de la columnaV1, los valores de la columna VX son las raíces cuadradas de los contenidos de la columnaV2 y los valores de la columna VY son las raíces cuadradas de los contenidos de la columnaV3.
TEMP = SQRT(V1[10:20])
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen las raíces cuadradas de los valores en las filas 10-20 de la columnaV1. Las otras celdas de TEMP están vacías.
TEMP = SQRT(V1[10:50]:V2)
Crea dos nuevas columnas denominadas TEMP y VX, cada una con valores en las filas 1 a 41 (las otras celdas están vacías). Los valores en la columna TEMP son las raíces cuadradas de los valores en las filas 10-50 de la columnaV1 y los valores en la columna VX son las raíces cuadradas de los valores en las filas 10-50 de la columna V2.

## Funciones relacionadas

Función	Descripción
DIV	Divide un rango de datos especificado por otro
MULT	Multiplica el contenido de dos rangos de datos
POW	Calcula un valor base elevado a la(s) potencia(s) exponencial(es) especificada(s)

---

## STAT



### Sintaxis

STAT(data [, keyword])



## Parámetros

data

Los valores numéricos para calcular los momentos para (es decir, medio, desviación estándar, desvío y curtosis). Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte el apartado "Parámetros de función de macro" del capítulo en esta guía para el producto IBM . Tiene que haber al menos tres valores en data.

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las opciones siguientes:

ALL - Realiza el cálculo en todas las celdas de data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de data

Para obtener más detalles sobre la utilización de palabras clave en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

## Descripción

STAT calcula los primeros cuatro momentos de los valores en el rango de datos especificado. El primer momento es el promedio. El segundo momento es la desviación estándar. El tercer momento es el desvío y el cuarto momento es curtosis.

El número de columnas que devuelve la función de macro STAT depende de palabra clave y el número de columnas endatos.

- Si se utiliza la palabra clave ALL (el valor predeterminado), los momentos se calculan sobre los valores en datos. Se devuelve una columna que contiene cuatro valores.
- Si se utiliza la palabra claveCOL, los momentos se calculan por separado para cada columna de entrada. Se devuelve una columna para cada columna de entrada, cada una de las cuales contiene cuatro valores.
- Si se utiliza la palabra claveROW, los momentos se calculan en cada fila de datos. STAT devuelve cuatro columnas. Se listan los momentos en todas las filas para cada fila de rango de datos de entrada.

## Ejemplos

```
TEMP = STAT(MERGE(1, 2, 3, 4, 5)) o TEMP = STAT(MERGE(1, 2, 3, 4, 5), ALL)
```

Crea una nueva columna denominada TEMP que contiene los valores 3, 1.58, 0 y -1.2.

TEMP = STAT(V1)
Crea una nueva columna denominada TEMP que contiene los cuatro primeros momentos de la columna V1.
TEMP = STAT(V1:V3)
Crea una nueva columna denominada TEMP que contiene los primeros cuatro momentos de la columna V1, V2 y V3.
TEMP = STAT(V1[10:20])
Crea una nueva columna denominada TEMP que contiene los primeros cuatro momentos de las celdas en las filas 10-20 de la columna V1.
TEMP = STAT(V1[1:5]:V4)
Crea una nueva columna denominada TEMP que contiene los primeros cuatro momentos de las celdas en las filas 1-5 de las columnas V1 a V4.
TEMP = STAT(V1:V3, COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los cuatro valores en la columna TEMP son momentos de la columna V1, los cuatro valores en la columna VX son los momentos de la columna V2 y los cuatro valores en la columna VY son los momentos de la columna V3.
TEMP = STAT(V1[1:5]:V3, COL) o TEMP = STAT(V1[1:5]:V3[1:5], COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene cuatro valores. Los valores en la columna TEMP son los momentos de las celdas en las filas 1-5 de la columna V1, los valores en la columna VX son los momentos de las celdas en las filas de la columna V2, y los valores en la columna VY son los momentos de las celdas en las filas 1-5 de la columna V3.
TEMP = STAT(V1:V3, ROW)
Crea cuatro columnas nuevas denominadas TEMP, VX, VY y VZ. TEMP contiene el promedio de cada fila de las columnas V1, V2 y V3, VX contiene la desviación estándar, VY contiene el desvío y VZ contiene la curtosis.
TEMP = STAT(V1[50:100]:V3, ROW) o TEMP = STAT(V1[50:100]:V3[50:100], ROW)
Crea cuatro nuevas columnas denominadas TEMP, VX, VY y VZ cada una de las cuales contiene 51 filas. TEMP contiene el promedio, VX contiene la desviación estándar, VY contiene el desvío y VZ contiene la curtosis. La primera fila corresponde a la fila 50 de las columnas V1, V2 y V3. La segunda fila corresponde a la fila 51 y así sucesivamente.

## Funciones relacionadas

Función	Descripción
AVG o MEAN	Calcula la media aritmética o el promedio de un rango de celdas
KURTOSIS	Calcula la curtosis de un rango de celdas
SKEW	Calcula el desvío de la distribución de un rango de celdas
STDV o STDEV	Calcula la desviación estándar de un rango de celdas
VARIANCE	Calcula la varianza de un rango de celdas

---

## STDV o STDEV

### Sintaxis

STDV(data [, keyword clave]) STDEV(data [, keyword])

### Parámetros

data

Los valores numéricos para calcular la desviación estándar de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las opciones siguientes:

ALL - realiza el cálculo para todas la celdas en data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de data

Para obtener más detalle sobre la utilización de palabras clave en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

**Nota:** Muchas funciones de macro toman parámetros de palabra clave {ALL | COL | ROW}. Estas palabras clave no se aplican en **IBM Campaign** porque los datos de entrada son siempre una columna o un campo único. La macro siempre se comportará como si se especificara la palabra clave COL. Por lo tanto, no necesita especificar estas palabras claves cuando utilice **IBM Campaign** .

### Descripción

STDV calcula la desviación estándar de todas las celdas en el rango de datos especificado. La desviación estándar de una distribución es la raíz cuadrada de la varianza. La desviación estándar se calcula de la siguiente manera:

$$\sqrt{\frac{1}{n-1} \sum_{j=1}^n (x_j - \text{mean})^2}$$

donde  $lasx$ 's son muestras,  $n$  es el número de muestras y *promedio* es el promedio de la distribución.

**Nota:** Si el número de muestras  $n = 1$ , STDV devuelve el error.

Este macro está disponible en IBM Interact.

## Ejemplos

TEMP = STDV(MERGE(1, 2, 1, 0)) o TEMP = STDEV(MERGE(1, 2, 1, 0))
Crea una nueva columna denominada TEMP que contiene el valor 0.71.
TEMP = STDV(V1)
Crea una nueva columna denominada TEMP que contiene un valor único el cual es la desviación estándar de los contenidos de la columna V1.
TEMP = STDV(V1:V3)
Crea una nueva columna denominada TEMP que contiene un valor único el cual es la desviación estándar de los contenidos de las columnas V1, V2 y V3.
TEMP = STDV(V1[1:5]:V4)
Crea una nueva columna denominada TEMP que contiene un valor único el cual es la desviación estándar de las celdas en las filas 1-5 de las columnas V1 a V4.
TEMP = STDV(V1:V3, COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY. El valor único de la columna TEMP es la desviación estándar de los contenidos de la columna V1, el valor único en la columna VX es la desviación estándar de los contenidos de la columna V2 y el valor único en la columna VY es la desviación estándar de los contenidos de la columna V3.
TEMP = STDV(V1[10:50]:V3,COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene un valor único. El valor en la columna TEMP es la desviación estándar de las celdas en las filas 10-50 de la columna V1, el valor en la columna VX es la desviación estándar de las celdas en las filas de la columna V2 y el valor en la columna VY es la desviación estándar de las celdas en las filas 10-50 de la columna V3.
TEMP = STDV(V1:V3, ROW)
Crea una nueva columna denominada TEMP donde cada entrada de celda es la desviación estándar de la fila correspondiente en las columnas V1, V2 y V3.
TEMP = STDV(V1[1:5]:V3,ROW)
Crea una nueva columna denominada TEMP, donde las celdas en las filas 1-5 contiene las desviaciones estándar de la fila correspondiente de las columnas V1 a V3. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
KURTOSIS	Calcula la curtosis de un rango de celdas
SKEW	Calcula el desvío de la distribución de un rango de celdas
STAT	Calcula desde el primer al cuarto momento del rango de datos especificado

Función	Descripción
VAR	Calcula la varianza de un rango de celdas

---

## STRING\_CONCAT

### Sintaxis

STRING\_CONCAT(string1, string2, ... stringN)

### Parámetros

string

Una cadena de texto ASCII a concatenar. Este puede ser un texto ASCII en comillas, una columna de texto, un rango de celdas que contiene texto o una expresión que evalúa algunos de los elementos anteriores. Consulte el apartado *Parámetros de la función de macro* del capítulo *Utilización de macros* para su producto para la definición de formato de cadena (igual que datos).

### Descripción

STRING\_CONCAT concatena los valores de texto ASCII en los rangos de datos especificados. Devuelve una nueva columna para cada entrada de columna, cada una contiene las cadenas concatenadas de las filas correspondientes de las cadenas. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** El ancho total de cada cadena resultante no puede exceder los 255 caracteres.

Este macro está disponible en IBM Interact.

IBM Interact también da soporte a la sintaxis siguiente:

STRING\_CONCAT( string1 , string2 , ... stringN )

Por ejemplo, STRING\_CONCAT('a', 'b', 'c', 'd') es válido.

### Ejemplos

TEMP = STRING_CONCAT("house", "boat")
Crea una nueva columna denominada TEMP, que contiene la cadena de texto ASCII "casa flotante".
TEMP = STRING_CONCAT(V1, ".")
Crea una nueva columna denominada TEMP, cada fila contiene la cadena de texto ASCII en la fila correspondiente de la columna V1 con un período añadido.
TEMP = STRING_CONCAT(V1, V2)
Crea una nueva columna denominada TEMP, cada fila contiene la cadena de texto ASCII en la columna V1 concatenada con la cadena de texto en la columna V2.

<pre>TEMP = STRING_CONCAT(V1:V3, V4:V6)</pre> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son las cadenas concatenadas de las filas correspondientes de la columna V1 y V4, los valores de la columna VX son las cadenas concatenadas de las filas correspondientes de la columna V2 y V5 y los valores de la columna VY son las cadenas concatenadas de las filas correspondientes de las columnas V3 y V6.</p>
<pre>TEMP = STRING_CONCAT(V1[5:10]:V2, V3:V4)</pre> <p>Crea dos nuevas columnas denominadas TEMP y VX. Los valores en la columna TEMP son las cadenas de las filas 5-10 de la columna V1 concatenadas con las filas 1-6 de la columna V3. Los valores en VX son las cadenas de las filas 5-10 de la columna V2 concatenadas con las filas 1-6 de la columna V4.</p>
<pre>TEMP = STRING_CONCAT('a', 'b', 'c', 'd')</pre> <p>Crea una nueva columna denominada TEMP, que contiene la cadena de texto ASCII "abcd".</p>

## Funciones relacionadas

Función	Descripción
STRING_HEAD	Devuelve los primeros n caracteres de cada cadena en el rango de datos especificado.
STRING_LENGTH	Devuelve la longitud de cada cadena en el rango de datos especificado
STRING_SEG	Devuelve el segmento de la cadena entre dos índices especificados
STRING_TAIL	Devuelve los últimos caracteres n de cada cadena en el rango de datos especificado

---

## STRING\_HEAD

### Sintaxis

```
STRING_HEAD(num_chars, data)
```

### Parámetros

num\_chars

el número de caracteres que se devuelve desde el comienzo de cada cadena en datos. Este debe ser un entero positivo mayor que cero.

data

Valores de cadena de texto ASCII. Este puede ser un texto ASCII en comillas, una columna de texto, un rango de celdas que contiene texto o una expresión que evalúa algunos de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM.

### Descripción

STRING\_HEAD devuelve los primeros caracteres num\_chars de cada valor de cadena en el rango de datos especificado. Si num\_chars es mayor que el número de

caracteres en una cadena de texto, los caracteres restantes se rellenan con el carácter null " \0 ".

## Ejemplos

<p>TEMP = STRING_HEAD(3, "15-ENE 1997")</p> <p>Crea una nueva columna denominada TEMP, la cual contiene la cadena de texto ASCII "JAN".</p>
<p>TEMP = STRING_HEAD(10, "Presión")</p> <p>Crea una nueva columna denominada TEMP, que contiene la cadena de texto ASCII "Presión".</p>
<p>TEMP = STRING_HEAD(5, V1)</p> <p>Crea una nueva columna denominada TEMP la cual contiene los cinco primeros caracteres de cada cadena en la columna V1.</p>
<p>TEMP = STRING_HEAD(1, V1:V3)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los primeros caracteres de las cadenas en las correspondientes filas de la columna V1, los valores de la columna VX son los primeros caracteres de las cadenas en las filas correspondientes de la columna V2 y los valores de la columna VY son los primeros caracteres de las cadenas en las filas correspondientes de la columna V3.</p>
<p>TEMP = STRING_HEAD(12, V4[1:50]:V6]</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los 12 primeros caracteres de las cadenas en las filas 1-50 de la columna V1, los valores de la columna VX son los 12 primeros caracteres de las cadenas en las filas 1-50 de la columna V2 y los valores de la columna VY son los 12 primeros caracteres de las cadenas en las filas 1-50 de la columna V3.</p>

## Funciones relacionadas

Función	Descripción
STRING_CONCAT	Concatena dos cadenas de texto de los rangos de datos especificados
STRING_LENGTH	Devuelve la longitud de cada cadena en el rango de datos especificado
STRING_SEG	Devuelve el segmento de la cadena entre dos índices especificados
STRING_TAIL	Devuelve los últimos caracteres n de cada cadena en el rango de datos especificado

---

## STRING\_LENGTH

### Sintaxis

STRING\_LENGTH(data)

Parámetros

data

Los valores de la cadena de texto ASCII para calcular la longitud de. Este puede ser un texto ASCII en comillas, una columna de texto, un rango de celdas que contiene texto o una expresión que evalúa algunos de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

STRING\_LENGTH devuelve la longitud de cada valor de cadena en el rango de datos especificado. Devuelve una nueva columna para cada columna de entrada, cada una contiene la longitud de la cadena de texto correspondiente.

**Nota:** Si STRING\_LENGTH se aplica a las columnas que contienen datos numéricos, devuelve ceros.

## Ejemplos

TEMP = STRING_LENGTH("cuatro")
Crea una nueva columna denominada TEMP que contiene el valor 4.
TEMP = STRING_LENGTH(4)
Crea una nueva columna denominada TEMP que contiene el valor 0.
TEMP = STRING_LENGTH(V1)
Crea una nueva columna denominada TEMP, donde cada valor es la longitud de la cadena en la fila correspondiente de la columna V1.
TEMP = STRING_LENGTH(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son las longitudes de las cadenas en las filas correspondientes de la columna V1, los valores de la columna VX son las longitudes de las cadenas en las filas correspondientes de la columna V2 y los valores de la columna VY son las longitudes de las cadenas en las filas correspondientes de la columna V3.
TEMP = STRING_LENGTH(V4[1:50]:V6]
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son las longitudes de las cadenas en las filas 1-50 de la columna V1, los valores de la columna VX son las longitudes de las cadenas en las filas 1-50 de la columna V2 y los valores de la columna VY son las longitudes de las cadenas en las filas 1-50 de la columna V3.

## Funciones relacionadas

Función	Descripción
STRING_CONCAT	Concatena dos cadenas de texto de los rangos de datos especificados
STRING_HEAD	Devuelve los primeros n caracteres de cada cadena en el rango de datos especificado.
STRING_SEG	Devuelve el segmento de la cadena entre dos índices especificados
STRING_TAIL	Devuelve los últimos caracteres n de cada cadena en el rango de datos especificado



---

## PROPIEDAD\_DE\_SERIE

### Sintaxis

STRING\_PROPER(data)

### Parámetros

data

Valor de la serie que se convertirá.

### Descripción

STRING\_PROPER convierte cada serie de valor en el rango de datos especificado al cambiar la primera letra o cualquier letra que sigue a un carácter o símbolo de espacio en blanco (distinto del subrayado) en mayúsculas y todos los otros caracteres a minúsculas. Devuelve una nueva columna para cada columna de entrada, cada uno que contiene la serie convertida en la columna de entrada correspondiente.

### Ejemplos

Temp = STRING_PROPER
----------------------

---

## STRING\_SEG

### Sintaxis

STRING\_SEG(from, to, data)

### Parámetros

from

El número de caracteres de desplazamiento desde el comienzo de la serie a comenzar a extraer la serie de segmento. Este debe ser un entero positivo mayor que cero y menor que a o STRING\_SEG devuelve una serie vacía.

to

El número de caracteres de desplazamiento desde el comienzo de la serie a detener la extracción la serie de segmento. Este debe ser un entero positivo mayor o igual a de. Si a equivale a de (y a es menor o igual que la longitud de la serie), se devuelve un carácter.

data

Valores de cadena de texto ASCII. Este puede ser un texto ASCII en comillas, una columna de texto, un rango de celdas que contiene texto o una expresión que evalúa algunos de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

STRING\_SEG devuelve el segmento de serie entre dos índices de cada valor de serie en el rango de datos especificado. Si de es mayor que la longitud de una serie, no se devuelve nada. Si to es mayor que la longitud de una serie, se devuelven todos los caracteres desde de.

## Ejemplos

<pre>TEMP = STRING_SEG(1, 6, "15-ENE 1997")</pre>
Crea una nueva columna denominada TEMP, que contiene la serie de texto ASCII " 15-ene ".
<pre>TEMP = STRING_SEG(5, 20, "presión")</pre>
Crea una nueva columna denominada TEMP, que contiene la serie de texto ASCII" segura ".
<pre>TEMP = STRING_SEG(5, 6, V1)</pre>
Crea una nueva columna denominada TEMP que contiene el quinto y el sexto carácter de casa serie en la columna V1.
<pre>TEMP = STRING_SEG(10, 20, V1:V3)</pre>
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son caracteres 10-20 de las series en las filas correspondientes de la columna V1, los valores de la columnaVX son caracteres 10-20 de las series en las correspondientes filas de la columnaV2 y los valores de la columna VY son los caracteres 10-20 de las series en las filas correspondientes de la columna V3.
<pre>TEMP = STRING_SEG(5, 10, V4[1:50]:V6]</pre>
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en las columnas TEMP son caracteres 5-10 de las series en las filas 1-50 de la columna V1, los valores de la columna VX son caracteres 5-10 de las series en las filas 1-50 de la columnaV2 y los valores de la columna VY son caracteres 5-10 de las series en las filas 1-50 de la columna V3.

## Funciones relacionadas

Función	Descripción
STRING_CONCAT	Concatena dos cadenas de texto de los rangos de datos especificados
STRING_HEAD	Devuelve los primeros n caracteres de cada cadena en el rango de datos especificado.
STRING_LENGTH	Devuelve la longitud de cada cadena en el rango de datos especificado
STRING_TAIL	Devuelve los últimos caracteres n de cada cadena en el rango de datos especificado

---

## STRING\_TAIL

### Sintaxis

```
STRING_TAIL(num_chars, data)
```

### Parámetros

num\_chars

El número de caracteres que se devuelve desde el final de cada serie en datos. Este debe ser un entero positivo mayor que cero.

data

Valores de cadena de texto ASCII. Este puede ser texto ASCII texto entre comillas, una columna de texto, un rango de celdas que contiene texto o una expresión que se evalúa en algunos de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de la guía para su producto IBM .

## Descripción

STRING\_TAIL devuelve los últimos caracteres num\_chars de cada valor de serie en el rango de datos especificado. Todos los valores de cadena se rellenan hasta la longitud de la cadena más larga con caracteres nulos " \0 ". Entonces el último num\_chars se devuelve para cada cadena. Si num\_chars es mayor que el número de caracteres en una cadena de texto, se devuelve toda la cadena de texto.

## Ejemplos

TEMP = STRING_TAIL(3, "JAN 15, 1997")
Crea una nueva columna denominada TEMP, que contiene la cadena de texto ASCII " 997 ".
TEMP = STRING_TAIL(10, "Presión")
Crea una nueva columna denominada TEMP, que contiene la cadena de texto ASCII "Presión".
TEMP = STRING_TAIL(5, V1)
Crea una nueva columna denominada TEMP que contiene los cinco últimos caracteres de cada serie en la columna V1.
TEMP = STRING_TAIL(1, V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los últimos caracteres de las series en las correspondientes filas de la columnaV1, los valores de la columna VX son los últimos caracteres de las series en las filas correspondientes de la columna V2 y los valores de la columna VY son los últimos caracteres de la series en las filas correspondientes de la columnaV3.
TEMP = STRING_TAIL(12, V4[1:50]:V6]
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son los 12 últimos caracteres de las series en las filas 1-50 de la columnaV1, los valores de la columna VX son los 12 últimos caracteres de las series en las filas 1-50 de la columna V2 y los valores de la columna VY son los 12 últimos caracteres de las series en las filas 1-50 de la columnaV3.

## Funciones relacionadas

Función	Descripción
STRING_CONCAT	Concatena dos cadenas de texto de los rangos de datos especificados
STRING_HEAD	Devuelve los primeros n caracteres de cada cadena en el rango de datos especificado.
STRING_LENGTH	Devuelve la longitud de cada cadena en el rango de datos especificado

Función	Descripción
STRING_SEG	Devuelve el segmento de la cadena entre dos índices especificados

## SUBSAMPLE



### Sintaxis

SUBSAMPLE(num\_samples, data)

### Parámetros

num\_samples

El número de muestras a extraer. Este debe ser un entero positivo menor que el número de celdas en el rango de datos especificado (es decir, la función de macro SUBSAMPLE no se puede utilizar para *aumentar* el número de puntos de datos por réplica).

data

Los valores de la muestra de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúe a cualquiera de los anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

SUBSAMPLE uniformemente muestrea el número solicitado de puntos de datos del rango de datos especificado. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene num\_samples filas de números uniformemente extraídos de la columna de entrada correspondiente. El primer valor de la fila y cada *n*-th valor de la fila se devuelve a partir de entonces de modo tal que se extraen un total de num\_samples.

**Nota:** Esta función de macro se puede utiliza para aumentar o disminuir el número de muestras.

### Ejemplos

```
TEMP = SUBSAMPLE(100, V1)
```

Crea una nueva columna denominada TEMP, la cual contiene 100 valores 100 muestreados uniformemente desde la columnaV1.

```
TEMP = SUBSAMPLE(50, V1:V3)
```

Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene 50 valores. Los valores en la columna TEMP son los valores de muestra de los contenidos de la columna V1, los valores de la columna VX son los valores de muestra de los contenidos de la columna V2 y los valores de la columna VY son los valores de muestra de los contenidos de la columna V3.

```
TEMP = SUBSAMPLE(5, V1[0:100])
```

Crea una nueva columna denominada TEMP con valores en las cinco primeras filas. Lo datos se muestrean uniformemente desde las filas 0-100 de la columna V1.

```
TEMP = SUBSAMPLE(250, V1[1:10]:V2)
```

Crea dos nuevas columnas denominadas TEMP y VX, cada una con valores en las primeras 250 filas. Los valores en la columna TEMP se muestrean uniformemente desde las filas 1-400 de la columna V1 y los valores en la columna VX se muestrean uniformemente desde las filas 1-400 de la columna V2.

## Funciones relacionadas

Función	Descripción
EXTRACT	Extrae filas de acuerdo a los valores de una columna de predicado
SAMPLE_RANDOM	Devuelve la(s) columna(s) de de las n celdas, cada una de las cuales contiene una muestra aleatoria del rango de datos especificado

---

## SUBSTITUTE (SUSTITUIR)



### Sintaxis

```
SUBSTITUTE(data, from_table, to_table)
```

### Parámetros

data

Los valores numéricos o valores de cadena a convertir. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

from\_table

Una columna que contiene valores que desea convertir. La longitud de la columna from\_table debe ser la misma que to\_table.

to\_table

Una columna que contiene los valores a convertir. La longitud de la columna to\_table debe ser la misma que from\_table.

### Descripción

SUBSTITUTE convierte los valores en datos con la utilización de las parejas de sustitución especificadas en from\_table y to\_table. Cada valor en datos encontrados en from\_table se sustituye con un valor en la fila correspondiente de to\_table.

SUBSTITUTE se puede utilizar para cambiar y los valores numéricos y los valores de la cadena. Siempre devuelve un rango de datos con las mismas dimensiones como datos.

**Nota:** Si utiliza SUBSTITUTE para convertir cadenas en números o viceversa, se deben convertir *todos los* valores en datos. Es decir, cada valor en datos debe aparecer en from\_table. De lo contrario, la fila resultante contiene ???.

## Ejemplos

TEMP = SUBSTITUTE(COLUMN(1,5,10),COLUMN(1), COLUMN(7))
Crea una nueva columna denominada TEMP,la cual contiene los valores 7, 5, 10 (el valor 1 se sustituye con 7).
TEMP = SUBSTITUTE(COLUMN("blue", "red"), COLUMN("blue", "red"), COLUMN(0, 1))
Crea una nueva columna denominada TEMP,que contiene los valores 0 y 1 (la cadena "azul" se reemplaza con 0 y "rojo" se sustituye con 1).
TEMP = SUBSTITUTE(V1, V2, V3)
Crea una nueva columna denominada TEMP que contiene los valores de la columna V1, donde los valores que se encuentran en la columna V2 se sustituyen con el valor en la fila correspondiente de la columna V3.
TEMP = SUBSTITUTE(V1:V2, V4, V5)
Crea dos nuevas columnas denominadas TEMP y VX, cada una con valores de las columnas V1 y V2 respectivamente, donde los valores que se encuentran en la columna V4 se sustituyen con el valor en la fila correspondiente de la columnaV5.
TEMP = SUBSTITUTE(V1[10:20]:V2, V4, V5)
Crea dos nuevas columnas denominadas TEMP y VX, cada una con valores de las filas 10-20 de las columnas V1 y V2 respectivamente, donde los valores que se encuentran en la columna V4 se sustituyen con el valor en la fila correspondiente de la columna V5.

## Funciones relacionadas

Función	Descripción
EXTRACT	Extrae filas de acuerdo a los valores de una columna de predicado
ISMEMBER	Comprueba un rango de entrada en una "tabla" de valores, devolviendo un uno si un valor está contenido en la tabla. De lo contrario, devuelve cero.

---

## SUBSTR o SUBSTRING

### Sintaxis

SUBSTR(string\_value, start\_pos[, nchars]) o SUBSTR(string\_value DE start\_pos[ PARA nchars]) SUBSTRING(string\_value, start\_pos[, nchars]) o SUBSTRING(string\_value DE start\_pos[ PARA nchars])

## Parámetros

`string_value`

La cadena de la que se tomará una subcadena.

`start_pos`

El carácter inicial del cual se extraerá la subcadena. .

*nchars*

El número de caracteres a extraer (debe ser mayor que o igual a 0). Si no se proporciona este valor, se extraen todos los caracteres restantes en `string_value`.

## Descripción

SUBSTR o SUBSTRING extrae caracteres *nchars* de la cadena y comienza por `start_pos`. Si se omite *nchars*, SUBSTR y SUBSTRING extraen caracteres de `start_pos` a través del final de la serie. Los espacios finales se truncan automáticamente.

Este macro está disponible en IBM Interact.

**Importante:** IBM Interact solamente da soporte a los siguientes formatos: SUBSTR(`string_value`, `start_pos`[, *nchars*]) o SUBSTRING(`string_value`, `start_pos`[, *nchars*])

## Ejemplos

SUBSTR SUBSTR Devuelve	("abcdef" FROM 1 FOR 2) ("abcdef",1,2) 'ab'
SUBSTR SUBSTR Devuelve	("abcdef" FROM -2 FOR 4) ("abcdef",-2,4) 'a'
SUBSTR SUBSTR Devuelve	("abcdef" FROM 3) ("abcdef",3) 'cdef'

---

## SUM

### Sintaxis

SUM (`data` [, `keyword`])

### Parámetros

`data`

Los valores numéricos para calcular la suma de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de `data`, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

`keyword`

Esta palabra clave opcional determina como se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las opciones siguientes:

ALL - Realiza el cálculo en todas las celdas de data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de data

Para obtener más detalles sobre la utilización de palabras clave en IBM Campaign, consulte “Especificaciones de formato” en la página 9.

Para obtener más detalles en la utilización de palabras clave en IBM PredictiveInsight, consulte “Especificaciones de formato” en la página 27.

**Nota:** Varias funciones de macro toman los parámetros de la palabra clave {ALL | COL | ROW}. Estas palabras clave no se aplican en **IBM Campaign** porque los datos de entrada son siempre una columna o un campo único. La macro siempre se comportará como si se especificara la palabra clave COL. Por lo tanto, no necesita especificar estas palabras claves cuando utilice **IBM Campaign** .

## Descripción

SUM calcula la suma de todas las celdas en el rango de datos especificado. Devuelve a un columna única.

**Nota:** SUM es la misma que la función de macro TOTAL.

Este macro está disponible en IBM Interact.

## Ejemplos

TEMP = SUM(3)Crea una nueva columna denominada TEMP que contiene el valor tres.
TEMP = SUM((COLUMN(3, 5, 1))
Crea una nueva columna denominada TEMP que contiene el valor nueve.
TEMP = SUM(V1)
Crea una nueva columna denominada TEMP que contiene un valor único el cual es la suma de los contenidos de la columna V1.
TEMP = SUM(V1:V3)
Crea una nueva columna denominada TEMP que contiene un valor único el cual es la suma de los contenidos de las columnas V1, V2 y V3.
TEMP = SUM(V1[1:5]:V4)
Crea una nueva columna denominada TEMP que contiene un valor único el cual es la suma de las celdas en las filas 10-20 de las columnas V1 a V4.
TEMP = SUM(V1:V3, COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY. El valor único en la columna TEMP es la suma del contenido de la columna V1, el valor único en la columna VX es la suma del contenido de la columna V2 y el valor único en la columna VY es la suma del contenido de la columna V3.



TEMP = SUM(V1[1:5]:V3, COL)
Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene un valor único. El valor en la columna TEMP es la suma de las celdas en las filas 1-5 de la columna V1, el valor en la columna VX es la suma de las celdas en las filas 1-5 de la columna V2 y el valor en la columna VY es la suma de las celdas en las filas 1-5 de la columna V3.
TEMP = SUM(V1:V3, ROW)
Crea una nueva columna denominada TEMP, donde cada entrada de celda es la suma de la fila correspondiente en las columnas V1, V2 y V3.
TEMP = SUM(V1[1:5]:V3, ROW)
Crea una nueva columna denominada TEMP, donde las celdas en las filas 1-5 contienen la suma de la fila correspondiente de las columnas V1 a V3. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
AVG o MEAN	Calcula la media aritmética o el promedio de un rango de celdas
AVG_DEV	Calcula la desviación promedio de un rango de celdas.

---

## TAN

### Sintaxis

TAN(data [, units\_keyword])

### Parámetros

data

Los valores numéricos para calcular la tangente de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

units\_keyword

Esta palabra clave opcional determina si los valores de entrada y los resultados se interpretan como grados o radianes. Seleccione una de las siguientes opciones:

RADIAN - Realiza los cálculos en radianes (valor predeterminado)

DEGREE - Realiza los cálculos en grados

Si no se especifica este parámetro, el valor predeterminado es radianes. (Para convertir de radianes a grados, divida por PI y multiplíquelo por 180.)

Para obtener más detalles en la utilización de palabras clave en IBM Campaign, consulte “Especificaciones de formato” en la página 9.

Para obtener más detalles en la utilización de palabras clave en IBM PredictiveInsight, consulte “Especificaciones de formato” en la página 27.

## Descripción

TAN calcula la tangente de los valores en el rango de datos especificado. Devuelve una nueva columna para cada columna de entrada cada una de las cuales contiene la tangente de números en la columna de entrada correspondiente.

## Ejemplos

<p><code>TEMP = TAN(PI/4)</code> o <code>TEMP = TAN(PI/4, 0)</code> o <code>TEMP = TAN(PI/4, RADIAN)</code></p> <p>Crea una nueva columna denominada TEMP que contiene el valor uno.</p>
<p><code>TEMP = TAN(V1)</code></p> <p>Crea una nueva columna denominada TEMP, donde cada valor es la tangente (en radianes) de los contenidos en la columna V1.</p>
<p><code>TEMP = TAN(V1:V3, 1)</code> o <code>TEMP = TAN(V1:V3, DEGREE)</code></p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son las tangentes de los contenidos de la columna V1, los valores de la columna VX son las tangentes de los contenidos de la columna V2 y los valores de la columna VY son las tangentes de los contenidos de la columna V3. Todos los valores se expresan en grados.</p>
<p><code>TEMP = TAN(V1[1:5]:V2)</code></p> <p>Crea dos columnas nuevas denominadas TEMP y VX, cada una con valores en las filas 1 a 5 (las otras celdas están vacías). Los valores en la columna TEMP son las tangentes de las filas correspondientes de la columna V1 y los valores en la columna VX son las tangentes de las filas correspondientes de la columna V2. Todos los valores se expresan en radianes.</p>

## Funciones relacionadas

Función	Descripción
ATAN	Calcula la arcotangente del contenido del rango de datos especificado
COS	Calcula el coseno del contenido del rango de datos especificado
COT	Calcula la cotangente del contenido del rango de datos especificado
SIN	Calcula el seno del contenido del rango de datos especificado
TANH	Calcula la tangente hiperbólica del contenido del rango de datos especificado

---

## TANH

### Sintaxis

`TANH(data [, units_keyword])`

## Parámetros

data

Los valores numéricos para calcular la tangente hiperbólica de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

units\_keyword

Esta palabra clave opcional determina si los valores de entrada y los resultados se interpretan como grados o radianes. Seleccione una de las opciones siguientes:

RADIAN - Realiza los cálculos en radianes (valor predeterminado)

DEGREE - Realiza los cálculos en grados

Si no se especifica este parámetro, el valor predeterminado es radianes. (Para convertir de radianes a grados, divida por PI y multiplíquelo por 180.)

Para obtener más detalles sobre la utilización de las palabras claves en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles en la utilización de palabras clave en IBM PredictiveInsight, consulte "Especificaciones de formato" en la página 27.

## Descripción

TANH calcula la tangente hiperbólica de los valores en el rango de datos especificado. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene la tangente hiperbólica de los números en la columna de entrada correspondiente. La tangente hiperbólica de un número se calcula de la siguiente manera:

$$\tanh(x) = \frac{\sinh(x)}{\cosh(x)}$$

**Nota:** Si el valor  $x$  es demasiado grande, se devuelve un error de desbordamiento. Esto sucede si  $\tanh(x)$  excede el máximo valor de coma flotante de 32 bits. Si  $\cosh(x)$  es cero, TANH devuelve el máximo valor de coma flotante de 32 bits.

## Ejemplos

TEMP = TANH(PI) o TEMP = TANH(PI, 0) o TEMP = TANH(PI, RADIAN)
Crea una nueva columna denominada TEMP que contiene el valor uno.
TEMP = TANH(V1)
Crea una nueva columna denominada TEMP, donde cada valor es la tangente hiperbólica (en radianes) de los contenidos de la columna V1.

<p>TEMP = TANH(V1:V3, 1) o TEMP = TANH(V1:V3, GRADO)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. Estos valores en la columna TEMP son las tangentes hiperbólicas de los contenidos de la columna V1, los valores de la columna VX son las tangentes hiperbólicas de los contenidos de la columna V2 y los valores de la columna VY son las tangentes hiperbólicas de los contenidos de la columna V3. Todos los valores se expresan en grados.</p>
<p>TEMP = TANH(V1[1:5]:V2)</p> <p>Crea dos nuevas columnas denominadas TEMP y VX, con valores en las filas 1-5 (las otras celdas están vacías). Los valores en la columna TEMP son las tangentes hiperbólicas de las filas correspondientes de la columna V1 y los valores en la columna VX son las tangentes hiperbólicas de las filas correspondientes de la columna V2. Todos los valores se expresan en radianes.</p>

## Funciones relacionadas

Función	Descripción
ATAN	Calcula la arcotangente del contenido del rango de datos especificado
COSH	Calcula el número coseno hiperbólico del contenido del rango de datos especificado
COT	Calcula la cotangente del contenido del rango de datos especificado
SINH	Calcula el seno hiperbólico del contenido del rango de datos especificado
TAN	Calcula la tangente del contenido del rango de datos especificado

---

## TO (A)



### Sintaxis

```
begin TO end begin...end
```

### Parámetros

```
begin
```

El número de comienzo en el rango para crear. Este puede ser un valor constante entero o una expresión que evalúa una constante entera.

```
end
```

El número final en el rango a crear. Este puede ser un valor constante entero o una expresión que evalúa una constante entera.

## Descripción

T0 crea una columna única que contiene los valores enteros que comienzan con begin y finalizan con end. Esta función de macro se utiliza para definir la variable de tiempo en funciones recursivas (vea la macro función INIT).

**Nota:** El operador T0 se puede abreviar con tres puntos(...).

## Ejemplos

TEMP = 1 T0 10 o TEMP = 1...10
Crea una nueva columna denominada TEMP que contiene los valores 1-10.
TEMP = 0 a -10
Crea una nueva columna denominada TEMP que contiene los valores 0 a 10.

## Funciones relacionadas

Función	Descripción
COLUMN	Crea columnas nuevas, concatenando verticalmente los valores de entrada en cada columna
MERGE	Crea un grupo de datos concatenando horizontalmente los valores de entrada

---

# TOTAL

## Sintaxis

TOTAL(data [, keyword])

## Parámetros

data

Los valores numéricos para calcular la suma de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las opciones siguientes:

ALL - Realiza el cálculo en todas las celdas de data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de datos

Para obtener más detalles sobre la utilización de las palabras claves en IBM Campaign, consulte “Especificaciones de formato” en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte “Especificaciones de formato” en la página 27.

**Nota:** Muchas funciones de macro toman parámetros de palabra clave {ALL | COL | ROW}. Estas palabras clave no aplican en **IBM Campaign** porque los datos de entrada son siempre una columna única o un campo. La macro siempre se comportará como si se especificara la palabra clave COL. Por lo tanto, no necesita especificar estas palabras claves cuando utilice **IBM Campaign**.

## Descripción

TOTAL calcula la suma de todas la celdas en el rango de datos especificado.

**Nota:** TOTAL es la misma que la función de macroSUM.

## Ejemplos

TEMP = TOTAL(3)	Crea una nueva columna denominada TEMP que contiene el valor tres.
TEMP = TOTAL((COLUMN(3, 5, 1)))	
	Crea una nueva columna denominada TEMP que contiene el valor nueve.
TEMP = TOTAL(V1)	
	Crea una nueva columna denominada TEMP que contiene un valor único el cual es la suma de los contenidos de la columna V1.
TEMP = TOTAL(V1:V3)	
	Crea una nueva columna denominada TEMP que contiene un valor único el cual es la suma de los contenidos de las columnas V1, V2 y V3.
TEMP = TOTAL(V1[1:5]:V4)	
	Crea una nueva columna denominada TEMP que contiene un valor único el cual es la suma de las celdas en las filas 10-20 de las columnas V1 a V4.
TEMP = TOTAL(V1:V3, COL)	
	Crea tres nuevas columnas denominadas TEMP, VX y VY. El valor único en la columna TEMP es la suma del contenido de la columna V1, el valor único en la columna VX es la suma del contenido de la columna V2 y el valor único en la columna VY es la suma del contenido de la columna V3.
TEMP = TOTAL(V1[1:5]:V3, COL)	
	Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene un valor único. El valor en la columna TEMP es la suma de las celdas en las filas 1-5 de la columna V1, el valor en la columna VX es la suma de las celdas en las filas 1-5 de la columna V2 y el valor en la columna VY es la suma de las celdas en las filas 1-5 de la columna V3.
TEMP = TOTAL(V1:V3, ROW)	
	Crea una nueva columna denominada TEMP donde cada entrada de celda es la suma de la fila correspondiente de las columnas V1, V2 y V3.

```
TEMP = TOTAL(V1[1:5]:V3, ROW)
```

Crea una nueva columna denominada TEMP, donde las celdas en las filas 1-5 contienen la suma de la fila correspondiente de las columnas V1 a V3. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
AVG o MEAN	Calcula la media aritmética o el promedio de un rango de celdas
AVG_DEV	Calcula la desviación promedio de un rango de celdas.

---

## TRANSPOSE (TRANSPONER)



### Sintaxis

TRANSPOSE (data)

### Parámetros

data

Los valores numéricos o valores de cadena a transponer. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

TRANSPOSE transpone el rango de datos especificado. Cambia la orientación horizontal y vertical del rango de datos (es decir, la primera fila de datos se convierte en la primera columna, la segunda fila se convierte en la segunda columna y así sucesivamente).

**Nota:** El rango de datos transpuesto debe ser rectangular. Las celdas vacías en una columna numérica se reemplazan por ceros. Las celdas vacías en una columna de cadena se reemplazan por la cadena vacía ("").

### Ejemplos

```
TEMP = TRANSPOSE(COLUMN(1,2,3))
```

Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene un valor único 1, 2 y 3.

```
TEMP = TRANSPOSE(MERGE("a", "b"))
```

Crea una nueva columna denominada TEMP que contiene a y b.

TEMP = TRANSPOSE(V1)
Crea una nueva columna para cada fila en la columna V1. Cada columna contiene un valor, el valor de la fila correspondiente de la columna V1.
TEMP = TRANSPOSE(V1:V3)
Crea una nueva columna para cada fila de la columna más larga, V1, V2 o V3. Cada columna tiene tres filas que contienen los valores transpuestos de V1:V3.
TEMP = TRANSPOSE(V1[10:15])
Crea seis nuevas columnas, cada una de la cuales contiene una fila. La primera columna contiene el valor de V1[10], la segunda columna contiene V1[11] y así sucesivamente.
TEMP = TRANSPOSE(V1[50:99]:V2)
Crea 100 nuevas columnas. Cada columna tiene dos filas que contienen los valores transpuestos de las filas 50-99 de las columnas V1 y V2.

## Funciones relacionadas

Función	Descripción
COLUMN	Crea columnas nuevas, concatenando verticalmente los valores de entrada en cada columna
MERGE	Crea un grupo de datos concatenando horizontalmente los valores de entrada

---

## TRUNCATE (TRUNCAR)

### Sintaxis

TRUNCATE (data)

### Parámetros

data

Los valores numéricos para truncar. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

### Descripción

TRUNCATE calcula el componente entero de cada valor del rango de datos especificado. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene el número entero (que no es fracción) parte de los números en la columna de entrada correspondiente.

**Nota:** La macro función FRACTION y la macro función TRUNCATE son complementarias porque suman a los valores originales.



## Ejemplos

TEMP = TRUNCATE(4.3)
Crea una nueva columna denominada TEMP que contiene el valor 4.
TEMP = TRUNCATE(2.9)
Crea una nueva columna denominada TEMP que contiene el valor -2.
TEMP = TRUNCATE(V1)
Crea una nueva columna denominada TEMP, donde cada valor es la parte fraccionaria del contenido de la columna V1.
TEMP = TRUNCATE(V1:V3)
Crea tres nuevas columnas denominadas TEMP, VX y VY. Los valores en la columna TEMP son las partes truncadas de la columna V1, los valores de la columna VX son las partes truncadas de la columna V2 y los valores de la columna VY son las partes truncadas de la columna V3.
TEMP = TRUNCATE(V1[10:20])
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen las partes truncadas de los valores en las filas 10-20 de la columna V1. Las otras celdas de TEMP están vacías.
TEMP = TRUNCATE(V1[50:99]:V2)
Crea dos nuevas columnas de nombre TEMP y VX, cada uno con valores en las filas 1-50 (las otras celdas están vacías). Los valores en la columna TEMP son las partes truncadas de las filas de la columna V1 y los valores en la columna VX son las partes truncadas de los valores en la columna V2.

## Funciones relacionadas

Función	Descripción
CEILING	Calcula el ceiling de cada valor del rango de datos especificado
FLOOR	Calcula el floor de cada valor del rango de datos especificado
FRACTION	Devuelve la parte fraccionaria de cada valor del rango de datos especificado

---

## UPPER

### Sintaxis

UPPER(data)

### Parámetros

data

El valor de la cadena a convertir en mayúscula.

## Descripción

UPPER convierte en mayúsculas cada valor de cadena en el rango de datos especificado. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene la cadena de mayúscula en la columna de entrada correspondiente.

Este macro está disponible en IBM Interact.

## Ejemplos

<pre>Temp = UPPER "gold"</pre>
Crea una nueva columna denominada Temp que contiene "ORO".
<pre>TEMP = UPPER( "jan 15, 1997")</pre>
Crea una nueva columna denominada TEMP, la cual contiene la cadena de texto ASCII "15-ene 1997".
<pre>TEMP = UPPER( "Pressure")</pre>
Crea una nueva columna denominada TEMP, la cual contiene la cadena de texto ASII "PRESIÓN".
<pre>TEMP = UPPER(V1)</pre>
Crea una nueva columna denominada TEMP la cual contiene caracteres en mayúscula para cada cadena en la columna V1.

---

## VARIANZA

### Sintaxis

VARIANCE(data [, keyword])

### Parámetros

data

Los valores numéricos para calcular la varianza de. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las opciones siguientes:

ALL - Realiza el cálculo en todas las celdas en data (valor predeterminado)

COL - Realiza el cálculo por separado para cada columna de data

ROW - Realiza el cálculo por separado para cada fila de data

Para obtener más detalles sobre la utilización de palabras clave en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

Para obtener más detalles sobre la utilización de palabras clave en IBM PredictiveInsight, consulte “Especificaciones de formato” en la página 27.

**Nota:** Muchas funciones de macro toman parámetros de palabra clave {ALL | COL | ROW}. Estas palabras clave no se aplican en **IBM Campaign** porque los datos de entrada son siempre una columna o un campo único. La macro siempre se comportará como si se especificara la palabra clave COL. Por lo tanto, no necesita especificar estas palabras claves cuando utilice **IBM Campaign** .

## Descripción

VARIANCE calcula la varianza de todos los valores en el rango de datos especificado. La varianza es el cuadrado de desviación estándar. La varianza se calcula de la siguiente manera:

$$\frac{1}{n - 1} \sum_{j=1}^n (x_j - \text{mean})^2$$

donde las  $x$  son las muestras,  $n$  es el número de muestras y *promedio* es el promedio de distribución.

**Nota:** Si el número de muestras  $n = 1$ , VARIANCE devuelve un error.

## Ejemplos

TEMP = VARIANCE(MERGE(3, 4, 5)) o TEMP = VARIANCE(MERGE(3, 4, 5), ALL)
Crea una nueva columna denominada TEMP que contiene el valor 0.67.
TEMP = VARIANCE(MERGE(-10, 5, 10))
Crea una nueva columna denominada TEMP que contiene el valor 72.2.
TEMP = VARIANCE(V1)
Crea una nueva columna denominada TEMP que contiene un valor único el cual es la varianza de los contenidos de la columna V1.
TEMP = VARIANCE(V1:V3)
Crea una nueva columna denominada TEMP que contiene un valor único el cual es la varianza de los contenidos de la columna V1, V2 y V3.
TEMP = VARIANCE(V1[10:20])
Crea una nueva columna denominada TEMP que contiene un valor único el cual es la varianza de las celdas en las filas 10-20 de la columna V1.
TEMP = VARIANCE(V1[1:5]:V4)
Crea una nueva columna denominada TEMP que contiene un valor único el cual es la varianza de las celdas en las filas 1-5 de las columnas V1 a V4.

<p>TEMP = VARIANCE(V1:V3, COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. El valor único en la columna TEMP es la varianza de los contenidos de la columna V1, el valor único en la columna VX es la varianza de los contenidos de la columna V2 y el valor único en la columna VY es la varianza de los contenidos de la columna V3.</p>
<p>TEMP = VARIANCE(MERGE(1,4), COL)</p> <p>Crea dos nuevas columnas denominadas TEMP y VX, cada una de las cuales contiene el valor cero.</p>
<p>TEMP = VARIANCE_(V1[1:5]:V3, COL) o TEMP = VARIANCE(V1[1:5]:V3[1:5], COL)</p> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY, cada una de las cuales contiene un valor único. El valor en la columna TEMP es la varianza de las celdas en las filas 1-5 de la columna V1, el valor en la columna VX es la varianza de las celdas en las filas 1-5 de la columna V2 y el valor en la columna VY es la varianza de las celdas en las filas 1-5 de la columna V3.</p>
<p>TEMP = VARIANCE(V1:V3, ROW)</p> <p>Crea una nueva columna denominada TEMP donde cada entrada de celda es la varianza de la fila correspondiente de las columnas V1, V2 y V3.</p>
<p>TEMP = VARIANCE(V1[1:5]:V3,ROW) o TEMP = VARIANCE(V1[1:5]:V3[1:5], ROW)</p> <p>Crea una nueva columna denominada TEMP, donde la celdas en las filas 1-5 contiene la varianza de la fila correspondiente de las columnas V1 a V3. Las otras celdas de TEMP están vacías.</p>

## Funciones relacionadas

Función	Descripción
KURTOSIS	Calcula la curtosis de un rango de celdas
SKEW	Calcula el desvío de la distribución de un rango de celdas

---

## WEEKDAY (DÍASEM)

### Sintaxis

WEEKDAY(data [, conversion\_keyword])

### Parámetros

data

Los datos de texto ASCII para convertir a valores numéricos representan los días de la semana (1-7). Este puede ser texto ASCII texto entre comillas, una columna de texto, un rango de celdas que contiene texto o una expresión que evalúa algunos de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

conversion\_keyword

Esta palabra clave opcional especifica cómo interpretar formatos de texto para fechas y horas. Seleccione una de las siguientes opciones:

1 - mm/dd/yy (valor predeterminado)

2 - dd-mmm-yy

3 - mm/dd/yy hh:mm

Si no se especifica este parámetro, el valor predeterminado es 1.

## Descripción

WEEKDAY convierte los valores de texto en el rango de datos especificado en valores numéricos que representan los días de la semana al utilizar el formato especificado para convertir fechas y horas. El número 0 para domingo, un 1 para lunes y así sucesivamente hasta 6 para el domingo. Si una cadena de texto no se puede analizar con la utilización de `conversion_keyword`, WEEKDAY devolverá un error.

## Ejemplos

<pre>TEMP = WEEKDAY("1/1/95")</pre> <p>Crea una nueva columna denominada TEMP que contiene el número 0 (1 de enero de 1995 es un domingo).</p>
<pre>TEMP = WEEKDAY(V1, 2)</pre> <p>Crea una nueva columna denominada TEMP que contiene los números para los días de la semana para las cadenas de texto en la columna V1. Se espera que todas las cadenas de texto en la columna V1 tengan el formato dd-mmm-yy (de lo contrario se devuelven???)</p>
<pre>TEMP = WEEKDAY(V1:V3, 3)</pre> <p>Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene números que representan los días de la semana de las cadenas de texto en la columna V1. La columna VX contiene los números que representan los días de la semana de las cadenas de texto en la columna V2. La columna VY contiene los números que representan los días de la semana de las cadenas de texto en la columna V3. Se espera que todas las cadenas de texto en las columnas V1 - V3 tengan el formato mm/dd/yy hh:mm (de lo contrario se devuelven ???).</p>
<pre>TEMP = WEEKDAY(V1[10:20]:V2, 10)</pre> <p>Crea dos nuevas columnas denominadas TEMP y VX. La columna TEMP contiene los números que representan los días de la semana de las cadenas de textos en las filas 10-20 de la columna V1. La columna VX contiene los números que representan los días de la semana de las cadenas de texto en las filas 10-20 de la columna V2. Se espera que todas las cadenas de texto tengan el formato mm/dd/yy (de lo contrario se devuelve ???).</p>

## Funciones relacionadas

Función	Descripción
NUMBER	Convierte las cadenas de texto ASCII para horas y fechas a valores numéricos.

---

## WEEKDAYOF

### Sintaxis

WEEKDAYOF(date\_string [, input\_format])

## Parámetros

date\_string

Un texto que representa una fecha válida.

input\_format

Una de las palabras claves en la tabla siguiente que especifica el formato de fecha de date\_string.

## Descripción

WEEKDAYOF devuelve el día de la semana como un número entre 0-6(Domingo 0, Lunes 1 y así sucesivamente) para el rango de datos especificado por el date\_string. Si no se proporciona input\_format, se utilizará la palabra clave predeterminada DELIM\_M\_D\_Y.

## Ejemplos

WEEKDAYOF("08312000", MMDDYYYY) devuelve el número 4, ya que el jueves es el cuarto día de la semana.

**Nota:** Consulte "DATE" en la página 77 para obtener más información sobre formatos de fecha válidos.

## Funciones relacionadas

Función	Descripción
DAYOF	Devuelve el día actual del mes en forma de número.
MONTHOF	Devuelve el mes del año como número.
YEAROF	Devuelve el año como número.

---

## XOR

### Sintaxis

```
data1 XOR data2
```

### Parámetros

data1

Los enteros no negativos para XOR a nivel de bit con los valores en data2. Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a uno de los elementos anteriores. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

data2

Los enteros no negativos para XOR a nivel de bit con los valores en data1. Este puede ser un valor constante, una columna, un rango de celdas o una expresión

que evalúa a uno de los elementos anteriores. El número de columnas en data2 debe ser igual al número de columnas en data1, a menos que data2 sea una constante. Para obtener la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para el producto de IBM .

## Descripción

XOR realiza un XOR a nivel de bit entre los dos rangos de datos especificados. Devuelve una nueva columna para cada columna de entrada, cada una de las cuales contiene la columna correspondiente en data1 unida por un XOR a nivel de bit a la columna correspondiente de data2 (es decir, la primera columna de data1 está unida por un XOR a nivel de bit a la primera columna de data, la segunda columna, a la segunda columna y así sucesivamente).

Si data2 es una constante, cada valor en data1 está unido por un XOR a nivel de bit por ese valor. Si data2 contiene una o más columnas, los cálculos se realizan fila por fila entre una columna de data1 y una columna de data2. La primera fila de data1 está unida por un XOR a nivel de bit al primer valor de fila de data2, la segunda fila, a la segunda fila y así sucesivamente. Este cálculo fila por fila produce un resultado para cada fila hasta el último valor de la columna más corta.

**Nota:** La precisión para esta función de macro se limita a los valores enteros menores que  $2^{24}$  . No se permiten valores negativos.

## Ejemplos

TEMP = 3 XOR 7
Crea una nueva columna denominada TEMP que contiene el valor cuatro (XOR a nivel de bit de 011 y 111 es igual a 100).
TEMP = V1 XOR 8
Crea una nueva columna de nombre TEMP, donde cada valor es el contenido de la columna V1 unido por un XOR a nivel de bit con el valor binario 1000.
TEMP = V1 XOR V1
Crea una nueva columna denominada TEMP que contiene todos los ceros (cada valor unido por un XOR a sí mismo produce cero).
TEMP = V1 XOR V2
Crea una nueva columna denominada TEMP, donde cada valor es el valor de fila de la columna V1 unido por un XOR a nivel de bit al valor de la fila correspondiente de la columna V2.
TEMP = V1:V3 XOR V4:V6
Crea tres nuevas columnas denominadas TEMP, VX y VY. La columna TEMP contiene los valores en V1 unidos por un XOR a nivel de bit a los valores de la fila correspondiente de la columna V4. La columna VX contiene los valores unidos por un XOR a nivel de bit de las columnas V2 y V5. La columna VY contiene los valores unidos por un XOR a nivel de bit de las columnas V3 y V6.
TEMP = V1[10:20] XOR V2 o TEMP = V1[10:20] XOR V2[1:11]
Crea una nueva columna denominada TEMP, donde las primeras 11 celdas contienen el resultado unido por el XOR a nivel de bit de los valores en las filas 10-20 de la columna V1 por los valores en las filas 1-11 de la columna V2. Las otras celdas de TEMP están vacías.

## Funciones relacionadas

Función	Descripción
BIT_AND	Calcula AND a nivel de bit entre dos rangos de datos especificados
BIT_NOT	Calcula NOT a nivel de bit del contenido del rango de datos especificado
BIT_OR	Calcula OR a nivel de bit entre dos rangos de datos especificados

---

## XTAB



### Sintaxis

XTAB(col1, col2 [, operator\_keyword, numeric\_col3])

### Parámetros

col1

La primera columna para producir el xtab de. Puede ser una constante, una columna, un rango de celdas de una sola columna o cualquier expresión que evalúa a una de las anteriores.

col2

La segunda para producir el xtab de. Puede ser una constante, una columna, un rango de celdas de una sola columna o cualquier expresión que evalúa a una de las anteriores.

operator\_keyword

Una de las palabras claves de operador válido (ver más abajo).

numeric\_col3

La tercera columna para producir el xtab de. Esta puede ser una constante, una columna, un rango de celdas de una única columna o una expresión que evalúa algunas de las anteriores y que contiene un valor numérico.

### Descripción

XTAB calcula valores distintos en col1 y col2. A continuación, calcula operator\_keyword de numeric\_col3 en la intersección de cada valor col1 con cada valor col2.

El operator\_keyword predetermina COUNTOF, en cuyo caso no se utiliza numeric\_col3.

Posible operator\_keywords son:

COUNTOF - devuelve el número de registros en cada intersección.



COUNTZERO - devuelve el número de registros en cada intersección para el cual numeric\_col3 es cero.

COUNTNONZERO - devuelve el número de registros en cada intersección para el cual numeric\_col3 no es cero.

COUNTNULL - devuelve el número de registros en cada intersección para el cual numeric\_col3 es NULL.

MINOF - devuelve el valor más pequeño de numeric\_col3 en cada intersección; devuelve el valor que falta si no hay valores en la intersección.

MAXOF - devuelve el valor mayor de numeric\_col3 en cada intersección; devuelve el valor que falta si no hay valores en la intersección.

SUMOF - devuelve la suma de todos los valores numeric\_col3 en cada intersección.

AVGOF - devuelve el promedio de todos los valores no NULL numeric\_col3 en cada intersección.

STDEVOF - devuelve la desviación estándar de todos los valores no NULL numeric\_col3 en cada intersección.

**Nota:** A la macro función XTAB le puede tomar un tiempo largo en calcular cuándo los datos son grandes. Un "cálculo..." se visualizará la barra de progreso hasta que se complete el cálculo. Si decide cancelar el cálculo, pulse en la "X" en la barra de progreso y suprima la definición de función que contiene la macro función XTAB.

## Ejemplos

```
TEMP=XTAB(V1,V2)
```

Crea una serie de filas y columnas que calculan el recuento de los valores distintos en las intersecciones de las columnas V1 y V2.

```
TEMP=XTAB(V4,V5, SUMOF V6)
```

Crea una serie de filas y columnas que representan la intersección de los valores distintos de las columnas V4 y V5. La medida en cada intersección es la suma de los valores en la columna V6 para las filas correspondientes a esa intersección.

---

## YEAROF

### Sintaxis

```
YEAROF(date_string [, input_format])
```

### Parámetros

date\_string

Un texto que representa una fecha válida.

input\_format

Una de las palabras claves en la tabla siguiente que especifica el formato de fecha de date\_string.

## Descripción

YEAROF devuelve el año como número para la fecha especificada por eldate\_string. Si no se proporciona input\_format, se utilizará la palabra clave predeterminada DELIM\_M\_D\_Y.

## Ejemplos

YEAROF("31082000", DDMMYYYY) devuelve el número 2000.

Para obtener más información sobre formatos de fecha válidos, consulte "DATE" en la página 77.

## Funciones relacionadas

Función	Descripción
DAYOF	Devuelve el día actual del mes en forma de número.
MONTHOF	Devuelve el mes del año como número.
WEEKDAYOF	Devuelve el día de la semana como número.

---

## Contacting IBM technical support

If you encounter a problem that you cannot resolve by consulting the documentation, your company's designated support contact can log a call with IBM technical support. Use the information in this section to ensure that your problem is resolved efficiently and successfully.

If you are not a designated support contact at your company, contact your IBM administrator for information.

### Information to gather

Before you contact IBM technical support, gather the following information:

- A brief description of the nature of your issue.
- Detailed error messages you see when the issue occurs.
- Detailed steps to reproduce the issue.
- Related log files, session files, configuration files, and data files.
- Information about your product and system environment, which you can obtain as described in "System information."

### System information

When you call IBM technical support, you might be asked to provide information about your environment.

If your problem does not prevent you from logging in, much of this information is available on the About page, which provides information about your installed IBM applications.

You can access the About page by selecting **Help > About**. If the About page is not accessible, you can obtain the version number of any IBM application by viewing the `version.txt` file located under the installation directory for each application.

### Contact information for IBM technical support

For ways to contact IBM technical support, see the IBM Product Technical Support website: ([http://www-947.ibm.com/support/entry/portal/open\\_service\\_request](http://www-947.ibm.com/support/entry/portal/open_service_request)).



---

## Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information about the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan Ltd.  
1623-14, Shimotsuruma, Yamato-shi  
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation  
170 Tracer Lane  
Waltham, MA 02451  
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

#### COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not

been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

---

## Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

---

## Privacy Policy and Terms of Use Considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. A cookie is a piece of data that a web site can send to your browser, which may then be stored on your computer as a tag that identifies your computer. In many cases, no personal information is collected by these cookies. If a Software Offering you are using enables you to collect personal information through cookies and similar technologies, we inform you about the specifics below.

Depending upon the configurations deployed, this Software Offering may use session and persistent cookies that collect each user's user name, and other personal information for purposes of session management, enhanced user usability, or other usage tracking or functional purposes. These cookies can be disabled, but disabling them will also eliminate the functionality they enable.

Various jurisdictions regulate the collection of personal information through cookies and similar technologies. If the configurations deployed for this Software Offering provide you as customer the ability to collect personal information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for providing notice and consent where appropriate.

IBM requires that Clients (1) provide a clear and conspicuous link to Customer's website terms of use (e.g. privacy policy) which includes a link to IBM's and Client's data collection and use practices, (2) notify that cookies and clear gifs/web beacons are being placed on the visitor's computer by IBM on the Client's behalf along with an explanation of the purpose of such technology, and (3) to the extent required by law, obtain consent from website visitors prior to the placement of cookies and clear gifs/web beacons placed by Client or IBM on Client's behalf on website visitor's devices

For more information about the use of various technologies, including cookies, for these purposes, See IBM's Online Privacy Statement at: <http://www.ibm.com/privacy/details/us/en> section entitled "Cookies, Web Beacons and Other Technologies."









Impreso en España