

Versión 9 Release 1.2
23 de septiembre de 2015

*Manual del usuario de IBM Macros for
IBM EMM*



Note

Before using this information and the product it supports, read the information in "Notices" en la página 165.

Esta edición se aplica a la versión 9, release 1, modificación 2 de la familia de productos de IBMCampaign y a todos los releases y modificaciones subsiguientes hasta que se indique lo contrario en nuevas ediciones.

© Copyright IBM Corporation 1998, 2015.

Contenido

Capítulo 1. Utilización de macros en IBM

Campaign	1
Resúmenes de funciones de macro para IBM Campaign	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 para IBM Campaign	9
Especificaciones de formato	9
Utilización de las constantes	10

Capítulo 2. Utilización de macros en

IBM Interact	11
Resúmenes de funciones de macro para IBM Interact	11
Funciones estadísticas	11
Funciones matemáticas y trigonométricas	12
Funciones de cadena	13
Funciones de fecha y hora	14
Funciones varias	15
Parámetros de función de macro para IBM Interact	15
Especificaciones de formato	15
Utilización de las constantes	16

Capítulo 3. IBM Referencia de macros 17

Formato de palabras clave válidas	17
Macro ABS	19
Macro ACOS	20
Macro ACOT	21
Macro ADD_MONTHS	23
Macro AND	24
Macro ASIN	26
Macro ATAN	27
Macro AVG	29
Macro BETWEEN	31
Macro BIT_AND	31
Macro BIT_NOT	33
Macro BIT_OR	34
Macro BIT_XOR	36
Macro CEILING	38
Macro COLUMN	39
Macro COS	40
Macro COSH	42
Macro COT	43
Macro COUNT	45
Macro CURRENT_DATE	46
Macro CURRENT_DAY	47
Macro CURRENT_JULIAN	48
Macro CURRENT_MONTH	48
Macro CURRENT_TIME	49
Configuración de la fecha en su aplicación web	49
Macro CURRENT_WEEKDAY	50
Macro CURRENT_YEAR	51

Macro DATE	51
Macro DATE_FORMAT	53
Macro DATE_JULIAN	54
Macro DATE_STRING	55
Macro DAY_BETWEEN	56
Macro DAY_FROMNOW	57
Macro DAY_INTERVAL	58
Macro DAYOF	59
Macro DISTANCE	59
Macro DIV	60
Macro EQ	62
Macro EXP	64
Macro EXTERNALCALLOUT	65
Macro FACTORIAL	66
Macro FLOOR	67
Macro FORMAT	68
Macro FRACTION	71
Macro GE	72
Macro GROUPBY	74
Macro GROUPBY_WHERE	77
Macro GT	78
Macro IF	80
Macro IN	81
Macro INT	82
Macro INVERSE	83
Macro IS	85
Macro ISERROR	85
Macro ISODD	86
Macro ISEVEN	87
Macro ISODD	88
Macro LE	89
Macro LIKE	91
Macro LN o LOG	93
Macro LOG2	94
Macro LOG10	95
Macro LOWER	97
Macro LT	97
Macro LTRIM	99
Macro MAX	99
Macro MEAN	101
Macro MIN	103
Macro MINUS	105
Macro MOD	106
Macro MONTHOF	108
Macro MULT	109
Macro NE	110
Macro NOT	112
Macro NUMBER	114
Macro OR	120
Macro POSITION	122
Macro PLUS	123
Macro POW	125
Macro RANDOM	126
Macro RANDOM_GAUSS	128
Macro ROUND	129
Macro ROWNUM	130

Macro RTRIM	131
Macro SIGN	131
Macro SIN	132
Macro SINH	134
Macro SQRT	135
Macro STDV o STDEV	136
Macro STRING_CONCAT	138
Macro STRING_HEAD	140
Macro STRING_LENGTH	141
Macro STRING_PROPER	142
Macro STRING_SEG	143
Macro STRING_TAIL	144
Macro SUBSTR o SUBSTRING	146
Macro SUM	146
Macro TAN	148
Macro TANH	149

Macro TOTAL	151
Macro TRUNCATE	153
Macro UPPER	154
Macro VARIANCE	155
Macro WEEKDAY	157
Macro WEEKDAYOF	158
Macro XOR	159
Macro YEAROF	160

Before you contact IBM technical support	163
---	------------

Notices	165
Trademarks	167
Privacy Policy and Terms of Use Considerations	167

Capítulo 1. Utilización de macros en IBM Campaign

Este capítulo proporciona información de uso sobre las macros de IBM® Campaign. Asegúrese de leer este capítulo antes de intentar utilizar macros en IBM Campaign.

Resúmenes de funciones de macro para IBM Campaign

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

Se proporcionan páginas de referencia detalladas para cada función de macro por orden alfabético en el Capítulo 3, “IBM Referencia de macros”, en la página 17.

Importante: Las macros se pueden aplicar a IBM Campaign y a IBM Interact, o a uno solo de estos productos. Las descripciones de macro identifican los productos en los que están disponibles.

Consulte “Parámetros de función de macro para IBM Campaign” en la página 9 para obtener 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
RECuento	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 número coseno del contenido del rango de datos especificado
COSH	Una columna para cada columna de entrada	Calcula el número NOT lógico del contenido del rango de datos especificado 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

Nombre de la macro	Devuelve	Descripción
FRACTION	Una columna para cada columna de entrada	Devuelve la parte fraccionaria de cada valor del rango de datos especificado
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

Nombre de la macro	Devuelve	Descripción
RANDOM_GAUSS	Una columna con el número especificado de valores	Devuelve el número especificado de valores aleatorios de una distribución gaussiana
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

Nombre de la macro	Devuelve	Descripción
TRUNCATE	Una columna para cada columna de entrada	Devuelve la parte no fraccionaria de cada valor del rango de datos especificado
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

Nombre de la macro	Devuelve	Descripción
STRING_LENGTH	Una columna para cada columna de entrada	Devuelve la longitud de cada cadena en el rango de datos especificado
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

Nombre de la macro	Devuelve	Descripción
DATE_JULIAN	Una columna para cada columna de entrada	Devuelve la fecha juliana
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 de 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)

Nombre de la macro	Devuelve	Descripción
ISODD	Una columna para cada columna de entrada	Comprueba si los valores de entrada son impares (es decir, no divisibles por dos)
ROWNUM	Una columna para cada columna de entrada	Genera números secuenciales desde uno hasta el número de registros

Parámetros de función de macro para IBM Campaign

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 los cálculos en varios campos a la vez o en un subconjunto de filas.

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 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 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
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 en LAST_NAME con el prefijo "Mr.".

Capítulo 2. Utilización de macros en IBM Interact

Este capítulo proporciona información de uso sobre las macros de IBM Interact. Asegúrese de leer este capítulo antes de intentar utilizar macros en IBM Interact.

Ayudante de fórmulas y comprobación de sintaxis de expresión de macro

Importante: El cuadro de diálogo Ayudante de fórmulas, incluida la función Comprobación de sintaxis, valida actualmente las expresiones de macro en función de lo que IBM Campaign soporta. Sin embargo, IBM Interact admite únicamente un subconjunto de la funcionalidad de la macro de Campaign. Por lo tanto, debe asegurarse de que las macros y las palabras clave (como las palabras clave de formato de fecha) utilizadas para Interact estén admitidas. Busque notas relacionadas con Interact in the Macros Reference chapter of este manual.

Resúmenes de funciones de macro para IBM Interact

Las tablas de las secciones siguientes proporcionan descripciones detalladas de las macros específicas de IBM Interact.

Se proporcionan páginas de referencia detalladas para cada función de macro por orden alfabético en el Capítulo 3, “IBM Referencia de macros”, en la página 17.

Importante: Las macros se pueden aplicar a IBM Campaign y a IBM Interact, o a uno solo de estos productos. Las descripciones de macro identifican los productos en los que están disponibles.

“Parámetros de función de macro para IBM Interact” en la página 15 proporciona información sobre los parámetros de entrada de función de macro para Interact.

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 15

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 claveCOL; una columna con valor para cada fila para la palabra claveROW.	Calcula la media aritmética o el promedio de un rango de celdas

Nombre de la macro	Devuelve	Descripción
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
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

Nombre de la macro	Devuelve	Descripción
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	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

Nombre de la macro	Devuelve	Descripción
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
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	Añade el número especificado de meses y devuelve una fecha
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 en el rango del 1 al 31
CURRENT_MONTH	Una columna para cada columna de entrada	Devuelve el mes actual del año como un número en el rango del 1 al 12
CURRENT_WEEKDAY	Una columna para cada columna de entrada	Devuelve la semana actual del mes como un número en el rango 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 para IBM Interact

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 los cálculos en varios campos a la vez o en un subconjunto de filas.

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 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 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 también 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. IBM Referencia de macros

Esta sección describe cada macro disponible para utilizar en IBM Campaign, IBM Interact, o ambos. Las macros se listan en orden alfabético.

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.

Formato de palabras clave válidas

La tabla siguiente muestra las palabras clave para formatos válidos, con una descripción y un ejemplo de cada una de ellas.

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
YYMM	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
YYYYMM	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

Palabra clave	Descripción	Ejemplo(s)
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
MMMDYY	Mes de 3 letras, día de 2 dígitos y año de 2 dígitos	MAR3170
MMMDYYYY	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)

Macro ABS

La macro ABS solo está disponible en IBM Campaign.

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 columna V3.
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.

Macro ACOS

La macro ACOS solo está disponible en IBM Campaign.

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 su producto.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.

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 π . Si se utiliza la palabra clave DEGREE, ACOS devuelve valores en el rango de 0 a 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), 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 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 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 arcoseno 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

Macro ACOT

La macro ACOT solo está disponible en IBM Campaign.

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 cualquiera de ellos. 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

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.

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

Macro ADD_MONTHS

La macro ADD_MONTHS está disponible en IBM Campaign

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 (DDMMYYYYY, 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 input_format determina tanto el formato de la serie de entrada como el formato de la serie de salida.

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 (DDMMYYYYY) 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 <code>input_format</code> en <code>formato_salida</code> .

Macro AND

La macro AND está disponible en IBM Campaign and IBM Interact.

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 cualquiera de éstos. 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 .

`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 cualquiera de éstos. 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 la guía para su producto 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).

Ejemplos

TEMP = 1 AND 8 o TEMP = 1 && 8
Crea una nueva columna denominada TEMP que contiene el valor uno (cualquier número distinto de cero es tratado como un uno).
TEMP = V1 && 1
Crea una nueva columna denominada TEMP con el valor uno para cada valor de la columna V1.
TEMP = V1 && V1
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.
TEMP = V1 && V2
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.
TEMP = V1:V3 && V4:V6
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.
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 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 especificado

Función	Descripción
OR	Calcula el OR lógico entre dos rangos de datos especificados

Macro ASIN

La macro ASIN solo está disponible en IBM Campaign.

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 cualquiera de ellos. 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 .

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.

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 valor 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.
TEMP = ASIN(V1[10:20])
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.
TEMP = ASIN(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 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.

Funciones relacionadas

Función	Descripción
ACOS	Calcula el arcocoseno 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

Macro ATAN

La macro ATAN solo está disponible en IBM Campaign.

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 cualquiera de ellos. 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 .

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.

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 de $-\pi/2$ a $\pi/2$. Si se utiliza la palabra clave DEGREE, ATAN devuelve valores en el rango de -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 columna V1 (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 valores en las 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
ATAN	Calcula la arcotangente del contenido del rango de datos especificado
TAN	Calcula la tangente del contenido del rango de datos especificado

Macro AVG

La macro AVG está disponible en IBM Campaign and IBM Interact.

Sintaxis

AVG(data [, keyword])

Parámetros

data

Los valores numéricos para los que se va a calcular la media aritmética. Estos valores pueden ser un valor constante, una columna, un rango de celdas o una expresión que se evalúa como cualquiera ellos. 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 .

keyword

Esta palabra clave opcional determina como se realiza el cálculo sobre el rango de datos de entrada. Seleccione una de las palabras clave 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 endata

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

Nota: Varias funciones de macro toman los parámetros de palabra clave {ALL | COL | ROW}. Estas palabras clave no se aplican en **IBM Campaign** porque los datos de entrada son siempre una única columna o campo. La macro siempre se comporta como si se especificara la palabra clave COL. Por lo tanto, no necesita especificar estas palabras clave cuando utiliza **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 y dividiendo el resultado entre el número de celdas. El número de columnas que AVG devuelve 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 nueva columna contiene un valor (el promedio de todas las celdas en la columna de entrada correspondiente).
- Si keyword es ROW, AVG devuelve una nueva columna que contiene el promedio de todas las columnas 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.

Ejemplos

TEMP = AVG(V1)
Crea una columna denominada TEMP que contiene un único valor que es la media aritmética del contenido de la columna V1.
TEMP = AVG(V1:V3)
Crea una columna denominada TEMP que contiene un único valor que es la media aritmética del contenido de las columnas V1, V2 y V3.
TEMP = AVG(V1[10:20])
Crea una columna denominada TEMP que contiene un único valor que es la media aritmética de las celdas en las filas 10 a 20 de la columna V1.
TEMP = AVG(V1[1:5]:V4)
Crea una columna denominada TEMP que contiene un único valor que es la media aritmética de las celdas en las filas 1 a 5 de las columnas V1 a V4.
TEMP = AVG(V1:V3, COL)
Crea tres columnas nuevas denominadas TEMP, VX y VY. El único valor de la columna TEMP es la media aritmética del contenido de la columna V1. El único valor de la columna VX es la media aritmética del contenido de la columna V2. El único valor de la columna VY es la media aritmética del contenido de la columna V3.
TEMP = AVG(V1[1:5]:V3, COL)
Crea tres columnas nuevas denominadas TEMP, VX y VY, cada una de las cuales contiene un único valor. El valor de la columna TEMP es la media aritmética de las celdas en las filas 1 a 5 de la columna V1. El valor de la columna VX es la media aritmética de las celdas en las filas 1 a 5 de la columna V2. El valor de la columna VY es la media aritmética de las celdas en las filas 1 a 5 de la columna V3.

TEMP = AVG(V1, ROW)
Crea una columna denominada TEMP, que contiene los mismos valores que la columna V1 (la media aritmética de cualquier número es el propio número).
TEMP = AVG(V1:V3, ROW)
Crea una columna denominada TEMP donde cada entrada de celda es la media aritmética de la fila correspondiente en las columnas V1, V2 y V3.
TEMP = AVG(V1[1:5]:V3, ROW)
Crea una columna denominada TEMP, donde las celdas de las filas 1 a 5 contienen la media aritmética de la fila correspondiente en las columnas V1 a V3. Las demás celdas de TEMP están vacías.

Funciones relacionadas

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

Macro BETWEEN

La macro BETWEEN solo está disponible en IBM Campaign.

Sintaxis

value1 BETWEEN value2 AND value3

Parámetros

Equivalente a 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

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.

Macro BIT_AND

La macro BIT_AND está disponible en IBM Campaign and IBM Interact.

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 cualquiera de éstos. 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 .

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 cualquiera de éstos. 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 la guía para su producto 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 data2. 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

TEMP = 3 BIT_AND 7 o TEMP = 3 & 7
Crea una nueva columna denominada TEMP que contiene el valor tres (AND a nivel de bit de 011 y 111 equivale a 011).
TEMP = V1 & 8
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.

TEMP = V1 & V1
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).
TEMP = V1 & V2
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.
TEMP = V1:V3 & V4:V6
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.
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 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
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
BIT_XOR o XOR	Calcula XOR a nivel de bit entre dos rangos de datos especificados

Macro BIT_NOT

La macro BIT_NOT está disponible en IBM Campaign and IBM Interact.

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 la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para su producto. 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 constante 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

Macro BIT_OR

La macro BIT_OR está disponible en IBM Campaign and IBM Interact.

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 cualquiera de éstos. 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 .

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 cualquiera de éstos. 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 la guía para su producto 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 2²⁴ . 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, 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

Macro BIT_XOR

La macro BIT_XOR está disponible en IBM Campaign and IBM Interact.

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 cualquiera de éstos. 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 .

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 cualquiera de éstos. 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 la guía para su producto 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 dedata1 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 data2 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
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

Macro CEILING

La macro CEILING solo está disponible en IBM Campaign.

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 cualquiera de ellos. 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

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

Macro COLUMN

La macro COLUMN solo está disponible en IBM Campaign.

Sintaxis

COLUMN(data [, data]...) o (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 la guía para su producto 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 de data 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), 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.

Macro COS

La macro COS solo está disponible en IBM Campaign.

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 cualquiera de ellos. 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 .

`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 palabrasclave en IBM Campaign, consulte “Especificaciones de formato” en la página 9.

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

TEMP = COS(PI) o TEMP = COS(PI, 0) o TEMP = COS(PI, RADIAN)
Devuelve una nueva columna de nombre TEMP que contiene el valor único de -1.
TEMP = COS(90, 1) o TEMP = COS(90, DEGREE)
Devuelve una nueva columna de nombre TEMP que contiene el valor único de cero.
TEMP = COS(V1) o TEMP = COS(V1, 0) o TEMP = COS(V1, RADIAN)
Crea una nueva columna de nombre TEMP, donde cada valor es el coseno (en radianes) del contenido de la columna V1.
TEMP = COS(V1:V3, 1)
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.
TEMP = COS(V1[10:20])
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.
TEMP = COS(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 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 seno del contenido del rango de datos especificado
TAN	Calcula la tangente del contenido del rango de datos especificado

Macro COSH

La macro COSH solo está disponible en IBM Campaign.

Sintaxis

`COSH(data [, units_keyword])`

Parámetros

`data`

Los valores numéricos para calcular el coseno hiperbólico de. Esto puede ser un valor constante, una columna, un rango de celdas, o una expresión para evaluar cualquiera de las 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 .

`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 palabrasclave en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

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

<code>TEMP = COSH(0)</code> o <code>TEMP = COSH(0, 0)</code> o <code>TEMP = COSH(0, RADIAN)</code>
--

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 número 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

Macro COT

La macro COT solo está disponible en IBM Campaign.

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

cualquiera de ellos. 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 .

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 palabrasclave en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

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

TEMP = COT(90) o TEMP = COT(90, 0) o TEMP = COT(90, RADIAN)
Devuelve una nueva columna de nombre TEMP que contiene el valor -0.5.
TEMP = COT(0)
Devuelve una nueva columna de nombre TEMP que contiene el valor MAX_FLOAT_32.
TEMP = COT(V1, 1) o TEMP = COT(V1, DEGREE)
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 número 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

Macro COUNT

La macro COUNT solo está disponible en IBM Campaign.

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 cualquiera de ellos. 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

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
SUM o TOTAL	Calcula la suma de un rango de celdas

Macro CURRENT_DATE

La macro CURRENT_DATE está disponible en IBM Campaign and IBM Interact.

Sintaxis

CURRENT_DATE([format])

Parámetros

format

Una de las palabras clave en la siguiente tabla especificando el formato de fecha de date_string.

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

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: No todos los formatos disponibles en IBM Campaign están soportados por 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.

Macro CURRENT_DAY

La macro CURRENT_DAY está disponible en IBM Campaign and IBM Interact.

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 .

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.

Función	Descripción
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.

Macro CURRENT_JULIAN

La macro CURRENT_JULIAN solo está disponible en IBM Campaign.

Sintaxis

CURRENT_JULIAN()

Descripción

CURRENT_JULIAN() devuelve el número de juliano para la fecha actual (el número de días transcurridos desde el 1 de enero del año 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.

Macro CURRENT_MONTH

La macro CURRENT_MONTH está disponible en IBM Campaign and IBM Interact.

Sintaxis

CURRENT_MONTH()

Descripción

CURRENT_MONTH devuelve el mes actual del año como un número entre 1-12.

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.

Macro CURRENT_TIME

La macro CURRENT_TIME solo está disponible en IBM Campaign.

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 dateFormat 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: webappphome 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 webappphome/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 webappphome/webapp, copie la totalidad del árbol del directorio en_US en language_code (distinción entre mayúsculas y minúsculas).
3. En webappphome/webapp/WEB-INF/classes/resources, copie StaticMessages_en_US.properties a StaticMessages_language_code.properties. También copie ErrorMessage_en_US.properties a ErrorMessage_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.

Macro `CURRENT_WEEKDAY`

La macro `CURRENT_WEEKDAY` está disponible en IBM Campaign and IBM Interact.

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.

Ejemplos

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

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_YEAR	Devuelve el año actual como un número.

Macro CURRENT_YEAR

La macro CURRENT_YEAR está disponible en IBM Campaign and IBM Interact.

Sintaxis

CURRENT_YEAR()

Descripción

CURRENT_YEAR devuelve el año actual en forma de número.

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.

Macro DATE

La macro DATE está disponible en IBM Campaign and IBM Interact.

Sintaxis

DATE(date_string [, format])

Parámetros

date_string

Texto que representa una fecha válida.

format

Una de las palabras clave de la tabla de “Formato de palabras clave válidas” en la página 17, 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 1º 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_YY.

Los formatos de fecha son de anchura fija (por ejemplo, la fecha 28 de febrero de 1970 se representa como 02281970 en el formato MMDDAAAA), 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.

Nota: No todos los formatos disponibles en IBM Campaign están soportados por IBM Interact.

Ejemplos

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

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.

Función	Descripción
CURRENT_DATE	Devuelve la fecha actual en un formato especificado.

Macro DATE_FORMAT

La macro DATE_FORMAT está disponible en IBM Campaign and IBM Interact.

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

Nota: No todos los formatos disponibles en IBM Campaign están soportados por IBM Interact.

Ejemplos

DATE_FORMAT("012171", MMDDYY, MMDDYYYY) devuelve la cadena "01211971".

Nota: Consulte "Macro DATE" en la página 51 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.

Macro DATE_JULIAN

La macro DATE_JULIAN solo está disponible en IBM Campaign.

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.

Macro DATE_STRING

La macro DATE_STRING solo está disponible en IBM Campaign.

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 1º 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)

Código	Descripción
%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
%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 “Macro DATE” en la página 51 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.

Macro DAY_BETWEEN

La macro DAY_BETWEEN solo está disponible en IBM Campaign.

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 “Macro DATE” en la página 51 para obtener más información sobre formatos de fecha válidos.

Funciones relacionadas

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

Macro DAY_FROMNOW

La macro `DAY_FROMNOW` solo está disponible en IBM Campaign.

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.

Macro DAY_INTERVAL

La macro DAY_INTERVAL solo está disponible en IBM Campaign.

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.

Macro DAYOF

La macro DAYOF solo está disponible en IBM Campaign.

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 “Macro DATE” en la página 51 para obtener más información sobre formatos de fecha válidos.

Macro DISTANCE

La macro DISTANCE está disponible en IBM Campaign and IBM Interact.

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. Utilice siempre comas y espacios para separar los valores numéricos. Esto es necesario para que se puedan admitir idiomas que utilizan la coma como separador de decimales, como se muestra a continuación en el segundo ejemplo.

Ejemplos

`DISTANCE (18.529747, 73.839798, 18.533511, 73.8777995, MILES, 2)` devuelve el valor 2.50 Millas.

`DISTANCE (18,529747, 73,839798, 18,533511, 73,8777995, KILOMETERS, 1)` devuelve el valor 4,0 kilómetros.

Macro DIV

La macro DIV está disponible en IBM Campaign and IBM Interact.

Sintaxis

data DIV divisor data / divisor

Parámetros

data

Los valores numéricos por los que dividir. Puede ser un valor constante, una columna, un rango de celdas o una expresión que se evalúa como 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 cualquiera de ellos. 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 (/).

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 or 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

Macro EQ

La macro EQ está disponible en IBM Campaign and IBM Interact.

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 la guía para su producto 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 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 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).

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

Macro EXP

La macro EXP solo está disponible en IBM Campaign.

Sintaxis

EXP(data)

Parámetros

data

Los valores numéricos utilizados como exponentes del número natural, e . Este puede ser un valor constante, una columna, un rango de celdas o una expresión que evalúa a cualquiera de éstos. 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

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 nueva columna para cada columna de entrada, cada una de las cuales contiene el resultado de e^x para cada valor de x en la columna o columnas de entrada correspondientes. EXP es la inversa de la función de macro LN.

Nota: Si el valor de x es demasiado grande o demasiado pequeño, se devuelve un error de 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 columna nueva 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 el resultado de elevar e al contenido de la columna V1, los valores de la columna VX son el resultado de elevar e al contenido de la columna V2 y los valores de la columna VY son el resultado de elevar e al contenido de la columna V3.
TEMP = EXP(V1[10:20])
Crea una columna nueva 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 el resultado de elevar e a los valores de la fila correspondiente de la columna V1, y los valores de la columna VX son el resultado de elevar e a los valores de fila correspondiente 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

Macro EXTERNALCALLOUT

La macro EXTERNALCALLOUT solo está disponible en IBM Interact.

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.

Macro FACTORIAL

La macro FACTORIAL solo está disponible en IBM Campaign.

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 una 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.

Macro FLOOR

La macro FLOOR solo está disponible en IBM Campaign.

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 la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para su producto.
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

Macro FORMAT

La macro FORMAT solo está disponible en IBM Campaign.

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

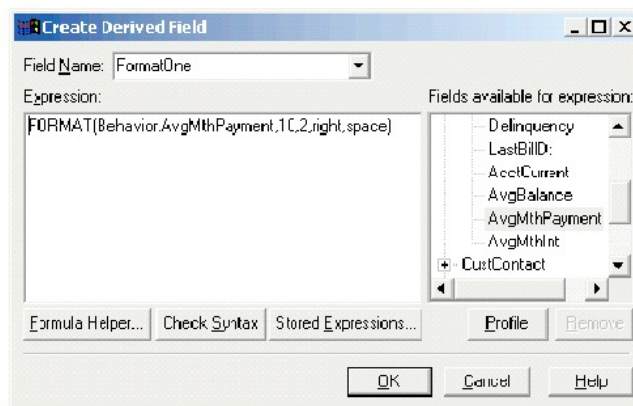
hecho, se convierten en necesarias). Por ejemplo: para alterar temporalmente la alineación a LEFT debe codificar: `FORMAT(myNumCol, 10, 2, PERIOD, LEFT)`.

Descripción

FORMAT convierte datos numéricos a una cadena formada de varios 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.



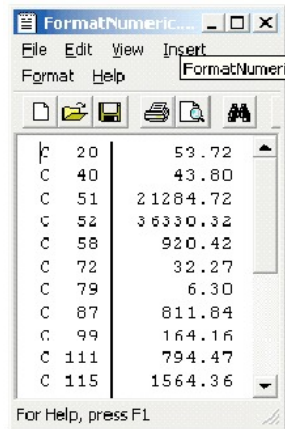
Los siguientes ejemplos muestran el mismo campo, AvgMthPayment, en tres formatos.

Sin formato:

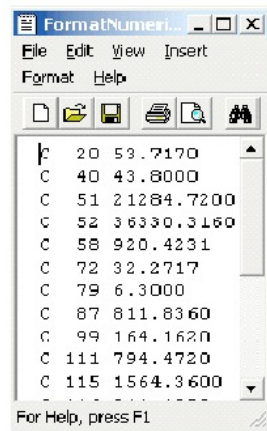
C	20	53.717
C	40	43.8
C	51	21284.72
C	52	36330.316
C	58	920.423123
C	72	32.271717
C	79	6.3
C	87	811.836
C	99	164.162
C	111	794.472
C	115	1564.36

For Help, press F1

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



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



Macro FRACTION

La macro FRACTION solo está disponible en IBM Campaign.

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 cualquiera de ellos. 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

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

Macro GE

La macro GE está disponible en IBM Campaign and IBM Interact.

Sintaxis

`data1 GE data2 data1 >= data2`

Parámetros

`data1`

El rango de celdas numéricas a comparar. Puede ser un valor constante, una columna, un rango de celdas o una expresión que se evalúa como 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 cualquiera de ellos. 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 la guía para su producto 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 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 así 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 GE puede ser abreviado con un signo mayor que seguido de un signo igual (>=).

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 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 todo 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
----	---

Macro GROUPBY

La macro GROUPBY solo está disponible en IBM Campaign.

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).
- keyword
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).

Palabra clave	¿Cadena? Si/No	Descripción
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.

Palabra clave	¿Cadena? Si/No	Descripción
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.

Macro GROUPBY_WHERE

La macro GROUPBY_WHERE solo está disponible en IBM Campaign.

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).
- keyword
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 "Macro GROUPBY" en la página 74 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.

Macro GT

La macro GT está disponible en IBM Campaign and IBM Interact.

Sintaxis

data1 GT data2 data1 > data2

Parámetros

data1

El rango de celdas numéricas a comparar. Puede ser un valor constante, una columna, un rango de celdas o una expresión que se evalúa como 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 cualquiera de ellos. 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 la guía para su producto 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 (>).

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 todo 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

Macro IF

La macro IF está disponible en IBM Campaign and IBM Interact.

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.

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 para IBM Campaign" 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 para IBM Campaign" 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, deseará crear una columna de predicado utilizando una de las funciones de macro de comparación (por ejemplo, ==, >, <, ISEVEN, ISODD, etc.).

Ejemplos

TEMP = IF(1, V1)	Crea una nueva columna denominada TEMP que contiene una copia de la columna V1.
TEMP = IF(V1, 1, 0)	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.
TEMP = IF(V3, V1, V2)	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.
TEMP = IF(ABS(V1-AVG(V1)) < STDV(V1), V1)	Crea una nueva columna denominada TEMP que contiene cada valor de la columna V1 que esté menos de una desviación estándar separado del promedio.
TEMP = IF(V3[20:30], V1[30:40], V2)	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).

Macro IN

La macro IN está disponible en IBM Campaign and IBM Interact.

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: 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.

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

Ejemplos

<code>TEMP = IN(25, COLUMN(1...10))</code>
Devuelve la(s) columna(s) especificada(s) de un rango de datos
<code>TEMP = IN("cat", COLUMN("cat", "dog", "bird"))</code>
Crea una nueva columna denominada TEMP que contiene el valor uno.
<code>TEMP = IN(V1, V1)</code>
Crea una nueva columna denominada TEMP que contiene todos los unos.
<code>TEMP = IN(V1, V2)</code>
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.

Macro INT

La macro INT solo está disponible en IBM Campaign.

Sintaxis

`INT(data)`

Parámetros

`data`

Los valores numéricos a redondear a un valor entero. Puede ser un valor constante, 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 la guía para su producto 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

<code>TEMP = INT(4.7)</code>
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

Macro INVERSE

La macro INVERSE solo está disponible en IBM Campaign.

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 cualquiera de ellos. 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

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

TEMP = INVERSE(3.2)
Crea una nueva columna denominada TEMP que contiene el valor -3.2.
TEMP = INVERSE(V1)
Crea una nueva columna denominada TEMP, donde cada valor es el valor negativo de los valores en la columna V1.
TEMP = INVERSE(V1:V3)
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.
TEMP = INVERSE(V1[10:20])
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.
TEMP = INVERSE(V1[1:5]:V2)
Crea dos columnas 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

Macro IS

La macro IS está disponible en IBM Campaign and IBM Interact.

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.

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.

Macro ISERROR

La macro ISERROR solo está disponible en IBM Campaign.

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 la función de macro" en el capítulo de esta guía para su producto 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.

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.

Macro ISODD

La macro ISODD solo está disponible en IBM Campaign.

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 la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para su producto. 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(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 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 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)

Macro ISEVEN

La macro ISEVEN solo está disponible en IBM Campaign.

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 cualquiera de ellos. 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

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(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, con valores en las 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)

Macro ISODD

La macro ISODD solo está disponible en IBM Campaign.

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 la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para su producto.

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(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 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 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)

Macro LE

La macro LE está disponible en IBM Campaign and IBM Interact.

Sintaxis

data1 LE data2 data1 <= data2

Parámetros

data1

El rango de celdas numéricas a comparar. Puede ser un valor constante, una columna, un rango de celdas o una expresión que se evalúa como 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 cualquiera de ellos. 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

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 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 data2, 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 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 (<=).

Ejemplos

TEMP = 4 LE 4 o TEMP = 4 <= 4
Crea una nueva columna denominada TEMP que contiene el valor uno (dado que cuatro es igual a sí mismo).
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 o 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 todo 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[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 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

Macro LIKE

La macro LIKE está disponible en IBM Campaign and IBM Interact.

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 la guía para su producto 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 la guía para su producto 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 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 data2, la segunda columna con la segunda columna, y así 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 compara 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.

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

Macro LN o LOG

La macro LN o LOG solo está disponible en IBM Campaign.

Sintaxis

LN(data) o LOG(data)

Parámetros

data

Los valores numéricos de los que calcular el logaritmo natural. Esto puede ser un valor constante, una columna, un rango de celdas, o una expresión para evaluar cualquiera de las 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

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, 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 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 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
POW	Calcula un valor base elevado a la(s) potencia(s) exponencial(es) especificada(s)

Macro LOG2

La macro LOG2 solo está disponible en IBM Campaign.

Sintaxis

LOG2(data)

Parámetros

data

Los valores numéricos de los que calcular el logaritmo en base 2. Puede ser un valor constante, 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 la guía para su producto 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

TEMP = LOG2(8)
Crea una nueva columna denominada TEMP que contiene el valor tres.
TEMP = LOG2(V1)
Crea una nueva columna denominada TEMP, donde cada valor es el logaritmo en base 2 del contenido de la columna V1.
TEMP = LOG2(V1:V3)
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.
TEMP = LOG2(V1[10:20])
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.
TEMP = LOG2(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 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

Macro LOG10

La macro LOG10 solo está disponible en IBM Campaign.

Sintaxis

LOG10(data)

Parámetros

data

Los valores numéricos de los que calcular el logaritmo en base 10. Puede ser un valor constante, 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 la guía para su producto 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, 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 100 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, con valores en las 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

Función	Descripción
LOG2	Calcula el logaritmo en base 2 del contenido del rango de datos especificado
POW	Potencia exponencial

Macro LOWER

La macro LOWER está disponible en IBM Campaign and IBM Interact.

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.

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.

Macro LT

La macro LT está disponible en IBM Campaign and IBM Interact.

Sintaxis

data1 LT data2 data1 < data2

Parámetros

data1

El rango de celdas numéricas a comparar. Puede ser un valor constante, una columna, un rango de celdas o una expresión que se evalúa como 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 cualquiera de ellos. 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

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" (<).

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

Macro LTRIM

La macro LTRIM está disponible en IBM Campaign and IBM Interact.

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.

Ejemplos

Temp = LTRIM " gold"

Crea una nueva cadena llamada Temp que contiene "gold".

Macro MAX

La macro MAX está disponible en IBM Campaign and IBM Interact.

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 cualquiera de ellos. 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 .

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo para 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 más detalles sobre el uso de palabras clave en IBM Campaign, vea "Especificaciones de formato" en la página 9.

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 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.

Ejemplos

TEMP = MAX(3) o TEMP = MAX(3, ALL)
Crea una nueva columna denominada TEMP que contiene el valor tres.
TEMP = MAX(V1)
Crea una nueva columna denominada TEMP que contiene un valor único que es el máximo valor de los contenidos de la columna V1.
TEMP = MAX(V1:V3)
Crea una nueva columna denominada TEMP que contiene un valor único que es el máximo de las columnas V1, V2 y V3.
TEMP = MAX(V1[10:20])
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.
TEMP = MAX(V1[1:5]:V4)
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.

TEMP = MAX(V1:V3, COL)
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.
TEMP = MAX(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 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.
TEMP = MAX(V1:V3, ROW)
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.
TEMP = MAX(V1[10:20]:V3, ROW)
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.

Funciones relacionadas

Función	Descripción
MIN	Calcula el número mínimo de un rango de celdas

Macro MEAN

La macro MEAN está disponible en IBM Campaign and IBM Interact.

Sintaxis

MEAN(data [, keyword])

Parámetros

data

Los valores numéricos de los que calcular la media aritmética. Puede ser un valor constante, 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 la guía para su producto 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 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

Consulte “Macro DATE” en la página 51 para obtener más detalles sobre la utilización de palabras clave.

Nota: Varias funciones de macro toman los 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 comportará como si se especificara la palabra clave COL. Por lo tanto, no es necesario especificar estas palabras clave al utilizar 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 nueva columna contiene un valor (el promedio de todas las celdas en la columna de entrada correspondiente).
- 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.

Ejemplos

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(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
SUM o TOTAL	Calcula la suma de un rango de celdas

Macro MIN

La macro MIN está disponible en IBM Campaign and IBM Interact.

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 cualquiera de ellos. 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 .

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 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

Consulte "Macro DATE" en la página 51 para obtener más detalles sobre la utilización de palabras clave.

Nota: Varias funciones de macro toman los 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 comportará como si se especificara la palabra clave COL. Por lo tanto, no es necesario especificar estas palabras clave al utilizar **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.

Ejemplos

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
MAX	Calcula el número máximo de un rango de celdas

Función	Descripción
MAX_TO_INDEX	Devuelve el índice de columna de valor máximo para cada fila de la columna especificada

Macro MINUS

La macro MINUS está disponible en IBM Campaign and IBM Interact.

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 cualquiera de ellos. 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 .

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 (-).

Ejemplos

TEMP = 7 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 or 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

Macro MOD

La macro MOD está disponible en IBM Campaign and IBM Interact.

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 la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para su producto.
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 cualquiera de ellos. 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 $TEMP = 5 \text{ MOD } 3$.

Ejemplos

$TEMP = 10 \text{ MOD } 8$ o $TEMP = 10 \% 8$ Crea una nueva columna denominada TEMP que contiene el valor 2.
$TEMP = -10 \% 8$ Crea una nueva columna denominada TEMP que contiene el valor -2.
$TEMP = V1 \% 8$ Crea una nueva columna denominada TEMP, donde cada valor es el contenido de la columna V1 módulo ocho.
$TEMP = V1:V3 \% 2$ 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.
$TEMP = V1 \% V1$ 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.
$TEMP = V1 \% V2$ 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 $V2=V1$, entonces se devuelven solo ceros, como en el ejemplo anterior.

TEMP = V1:V3 % V4:V6
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.
TEMP = V1[10:20] % V2 or 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 módulo 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
MOD	Calcula el módulo del contenido del rango de datos especificado

Macro MONTHOF

La macro MONTHOF solo está disponible en IBM Campaign.

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 "Macro DATE" en la página 51 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.

Macro MULT

La macro MULT está disponible en IBM Campaign and IBM Interact.

Sintaxis

```
data MULT multiplier data * multiplier
```

Parámetros

`data`

Los valores numéricos a multiplicar. Puede ser un valor constante, una columna, un rango de celdas o una expresión que se evalúa como 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 (*).

Ejemplos

TEMP = 8 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 or 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)

Macro NE

La macro NE está disponible en IBM Campaign and IBM Interact.

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 la guía para su producto 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 así 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 ($\neq$) o por un signo menor que seguido de un signo de mayor que ($<>$).

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 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

Macro NOT

La macro NOT está disponible en IBM Campaign and IBM Interact.

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 cualquiera de ellos. 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

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).

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 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 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

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

Macro NUMBER

La macro NUMBER está disponible en IBM Campaign y IBM Interact.

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 la función de macro" en el capítulo de esta guía para su producto 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)

La conversión de la palabra clave	Formato	Descripción
7	dd-mm	Convierte una fecha en el número de días desde el principio del año (por ejemplo, "01-FEB" a 32)
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)

La conversión de la palabra clave	Formato	Descripción
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	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)
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)

La conversión de la palabra clave	Formato	Descripción
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	aamdd	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	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)
37	aamdd	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	aaaamdd	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)

La conversión de la palabra clave	Formato	Descripción
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).

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 columnaV1. 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 columnaV3. Todas las fechas en el formato dd-mmm-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

Macro OR

La macro OR está disponible en IBM Campaign and IBM Interact.

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 la guía para su producto 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 cualquiera de éstos. 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 la guía para su producto 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).

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 todos los 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

Macro POSITION

La macro POSITION solo está disponible en IBM Campaign.

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.

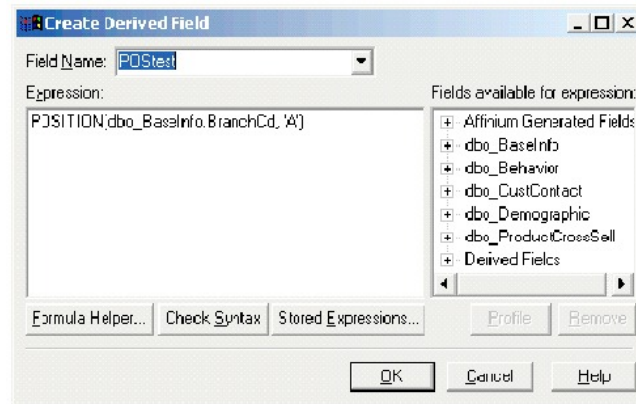
Descripción

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, se busca el patrón o cadena, 'A', dentro del valor de la columna, dbo_BaseInfo.BranchCd y se asigna 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.

C	AD
1	AD 1
2	AA 1
3	AC 1
4	BA 2
5	AD 1
6	AB 1
7	DE 0
8	BE 0
9	AE 1
10	BA 2
11	DC 0
12	BA 2

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".

Macro PLUS

La macro PLUS está disponible en IBM Campaign and IBM Interact.

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 la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para su producto.

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 cualquiera de ellos. 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 (+).

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 or 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

Macro POW

La macro POW está disponible en IBM Campaign and IBM Interact.

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 cualquiera de ellos. 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 cualquiera de éstos. 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 $TEMP = 2 \text{ POW } 8$.

Nota: Si el valor de x es demasiado grande o demasiado pequeño, se devuelve un desbordamiento. Esto ocurre si $\text{base}^{\text{exponent}}$ supera el máximo o mínimo de 32 bits del valor de coma flotante.

Ejemplos

$TEMP = 2 \text{ POW } 3$ o $TEMP = 2^3$ Crea una nueva columna denominada TEMP que contiene el valor ocho.
$TEMP = V1 \wedge 0.5$ Crea una nueva columna denominada TEMP, en donde cada valor es la raíz cuadrada del contenido de la columna V1 (equivalente a $\text{SQRT}(V1)$).
$TEMP = V1 \wedge V3$ 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.
$TEMP = V1:V3 \wedge V4:V6$ 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.
$TEMP = V1[10:20] \text{ POW } V2$ o $TEMP = V1[10:20] \text{ POW } V2[1:11]$ 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
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

Macro RANDOM

La macro RANDOM solo está disponible en IBM Campaign.

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

<code>TEMP = RANDOM()</code> Crea una nueva columna denominada <code>TEMP</code> que contiene números aleatorios de longitud ilimitada.
<code>TEMP = RANDOM(100)</code> Crea una nueva columna denominada <code>TEMP</code> que contiene 100 números aleatorios que se encuentran entre 0,0 y 1,0.
<code>TEMP = RANDOM(100, 5943049)</code> Crea una nueva columna denominada <code>TEMP</code> 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

Macro RANDOM_GAUSS

La macro RANDOM_GAUSS solo está disponible en IBM Campaign.

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

Macro ROUND

La macro ROUND solo está disponible en IBM Campaign.

Sintaxis

ROUND(data)

Parámetros

data

Los valores numéricos a redondear. Puede ser un valor constante, una columna, un rango de celdas o una expresión que se evalúa como 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

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

Macro ROWNUM

La macro ROWNUM solo está disponible en IBM Campaign.

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.

Macro RTRIM

La macro RTRIM está disponible en IBM Campaign and IBM Interact.

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.

Ejemplos

Temp = RTRIM "gold "

Crea una nueva cadena denominada Temp que contiene "gold".
--

Macro SIGN

La macro SIGN solo está disponible en IBM Campaign.

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 la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para su producto. 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(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

Macro SIN

La macro SIN solo está disponible en IBM Campaign.

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 "Macro Function Parameters" 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.

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

TEMP = SIN(PI/2) o TEMP = SIN(PI/2, 0) o TEMP = SIGN(PI/2, RADIAN)
Crea una nueva columna denominada TEMP que contiene el valor uno.
TEMP = SIN(V1)
Crea una nueva columna denominada TEMP, donde cada valor es el seno (expresado en radianes) del contenido de la columna V1.
TEMP = SIN(V1:V3, 1) o TEMP = SIN(V1:V3, DEGREE)
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.
TEMP = SIN(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 senos de los valores en las fila 10-50 de la columna V1 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.

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 número coseno 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

Macro SINH

La macro SINH solo está disponible en IBM Campaign.

Sintaxis

`SINH(data [, units_keyword])`

Parámetros

`data`

Los valores numéricos para calcular el seno hiperbólico de. Puede ser un valor constante, 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 la guía para su producto 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.

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

TEMP = SINH(1) o TEMP = SINH(1, 0) o TEMP = SINH(1, RADIAN)
Crea una nueva columna denominada TEMP que contiene el valor 1.18.
TEMP = SINH(V1)
Crea una nueva columna denominada TEMP, donde cada valor es un seno hiperbólico (en radianes) de los contenidos de la columna V1.
TEMP = SINH(V1:V3, 1) o TEMP = SINH(V1:V3, GRADO)
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.
TEMP = SINH(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 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.

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 seno del contenido del rango de datos especificado
TANH	Calcula la tangente hiperbólica del contenido del rango de datos especificado

Macro SQRT

La macro SQRT solo está disponible en IBM Campaign.

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 cualquiera de ellos. 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

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: Si 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 es 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 columna V1, los valores de la columna VX son las raíces cuadradas de los contenidos de la columna V2 y los valores de la columna VY son las raíces cuadradas de los contenidos de la columna V3.
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 columna V1. 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 columna V1 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)

Macro STDV o STDEV

La macro STDV o STDEV está disponible en IBM Campaign and IBM Interact.

Sintaxis

STDV(data [, keyword clave]) STDEV(data [, keyword])

Parámetros

data

Los valores numéricos para calcular la desviación estándar de. Esto puede ser un valor constante, una columna, un rango de celdas, o una expresión para evaluar cualquiera de las 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 .

keyword

Esta palabra clave opcional determina cómo se realiza el cálculo para 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 detalle sobre la utilización de palabras clave en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

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 es necesario especificar estas palabras clave al utilizar **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 las x '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.

Ejemplos

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
VAR	Calcula la varianza de un rango de celdas

Macro STRING_CONCAT

La macro STRING_CONCAT está disponible en IBM Campaign and IBM Interact.

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.

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.
TEMP = STRING_CONCAT(V1:V3, V4:V6)
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.
TEMP = STRING_CONCAT(V1[5:10]:V2, V3:V4)
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.
TEMP = STRING_CONCAT('a', 'b', 'c', 'd')
Crea una nueva columna denominada TEMP, que contiene la cadena de texto ASCII "abcd".

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

Función	Descripción
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

Macro STRING_HEAD

La macro STRING_HEAD solo está disponible en IBM Campaign.

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 la definición de formato de data, consulte la sección "Parámetros de función de macro" en el capítulo de esta guía para su producto. 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

TEMP = STRING_HEAD(3, "15-ENE 1997")
Crea una nueva columna denominada TEMP, la cual contiene la cadena de texto ASCII "JAN".
TEMP = STRING_HEAD(10, "Presión")
Crea una nueva columna denominada TEMP, la cual contiene la cadena de texto ASCII "Presión".
TEMP = STRING_HEAD(5, V1)
Crea una nueva columna denominada TEMP la cual contiene los cinco primeros caracteres de cada cadena en la columna V1.

TEMP = STRING_HEAD(1, V1:V3)
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.
TEMP = STRING_HEAD(12, V4[1:50]:V6]
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.

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

Macro STRING_LENGTH

La macro STRING_LENGTH solo está disponible en IBM Campaign.

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 la guía para su producto 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

Macro STRING_PROPER

La macro STRING_PROPER solo está disponible en IBM Campaign.

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

Macro STRING_SEG

La macro STRING_SEG solo está disponible en IBM Campaign.

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 alguno 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

TEMP = STRING_SEG(1, 6, "15-ENE 1997")
Crea una nueva columna denominada TEMP, que contiene la serie de texto ASCII " 15-ene ".
TEMP = STRING_SEG(5, 20, "presión")
Crea una nueva columna denominada TEMP, que contiene la serie de texto ASCII " segura ".
TEMP = STRING_SEG(5, 6, V1)
Crea una nueva columna denominada TEMP que contiene el quinto y el sexto carácter de casa serie en la columna V1.
TEMP = STRING_SEG(10, 20, V1:V3)
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 columna VX son caracteres 10-20 de las series en las correspondientes filas de la columna V2 y los valores de la columna VY son los caracteres 10-20 de las series en las filas correspondientes de la columna V3.
TEMP = STRING_SEG(5, 10, V4[1:50]:V6]
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 columna V2 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

Macro STRING_TAIL

La macro STRING_TAIL solo está disponible en IBM Campaign.

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 debe ser un texto ASCII en comillas, una columna de texto, un rango de celdas que contenga texto o una expresión que evalúe algunos de los elementos anteriores de las 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, la cual 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
STRING_SEG	Devuelve el segmento de la cadena entre dos índices especificados

Macro SUBSTR o SUBSTRING

La macro SUBSTR o SUBSTRING está disponible en IBM Campaign and IBM Interact.

Sintaxis

SUBSTR(string_value, start_pos[, nchars]) o SUBSTR(string_value FROM start_pos[FOR nchars]) SUBSTRING(string_value, start_pos[, nchars]) o SUBSTRING(string_value FROM start_pos[FOR nchars])

Parámetros

string_value

La cadena de la que se tomará una subcadena.

start_pos

El carácter inicial del cual se extraerá cada 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. Para evitar errores de sintaxis, asegúrese de que los valores numéricos están separados por comas y espacios, como se muestra en los ejemplos.

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'

Macro SUM

La macro SUM está disponible en IBM Campaign and IBM Interact.

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 "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 para el rango de datos de entrada. Seleccione una de las opciones siguientes:

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

ReCOL - 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 en la utilización de palabras clave en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

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

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.

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

Macro TAN

La macro TAN solo está disponible en IBM Campaign.

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 cualquiera de ellos. 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 .

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 en la utilización de palabras clave en IBM Campaign, consulte “Especificaciones de formato” en la página 9.

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

TEMP = TAN(PI/4) o TEMP = TAN(PI/4, 0) o TEMP = TAN(PI/4, RADIAN)
Crea una nueva columna denominada TEMP que contiene el valor uno.
TEMP = TAN(V1)
Crea una nueva columna denominada TEMP, donde cada valor es la tangente (en radianes) de los contenidos en la columna V1.
TEMP = TAN(V1:V3, 1) o TEMP = TAN(V1:V3, DEGREE)
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.
TEMP = TAN(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 las tangente 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.

Funciones relacionadas

Función	Descripción
ATAN	Calcula la arcotangente del contenido del rango de datos especificado
COS	Calcula el número 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

Macro TANH

La macro TANH solo está disponible en IBM Campaign.

Sintaxis

TANH(data [, units_keyword])

Parámetros

data

Los valores numéricos para calcular la tangente hiperbólica de. Esto puede ser un valor constante, una columna, un rango de celdas, o una expresión para evaluar cualquiera de las 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 .

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.

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.

TEMP = TANH(V1:V3, 1) o TEMP = TANH(V1:V3, GRADO)
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.
TEMP = TANH(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 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.

Funciones relacionadas

Función	Descripción
ATAN	Calcula la arcotangente 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
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

Macro TOTAL

La macro TOTAL está disponible en IBM Campaign and IBM Interact.

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 "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 para el rango de datos de entrada. Seleccione una de las opciones siguientes:

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

ReCOL - 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.

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

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

Macro TRUNCATE

La macro TRUNCATE solo está disponible en IBM Campaign.

Sintaxis

TRUNCATE (data)

Parámetros

data

Los valores numéricos para truncar. Puede ser un valor constante, una columna, un rango de celdas o una expresión que se evalúa como 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

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

Macro UPPER

La macro UPPER está disponible en IBM Campaign and IBM Interact.

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.

Ejemplos

Temp = UPPER "gold"
Crea una nueva columna denominada Temp que contiene "ORO".

TEMP = UPPER("jan 15, 1997")
Crea una nueva columna denominada TEMP, la cual contiene la cadena de texto ASCII "15-ene 1997".
TEMP = UPPER("Pressure")
Crea una nueva columna denominada TEMP, la cual contiene la cadena de texto ASCII "PRESIÓN".
TEMP = UPPER(V1)
Crea una nueva columna denominada TEMP la cual contiene caracteres en mayúscula para cada cadena en la columna V1.

Macro VARIANCE

La macro VARIANCE solo está disponible en IBM Campaign.

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 cualquiera de ellos. 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 .

keyword

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

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

ReCOL - 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 en la utilización de palabras clave en IBM Campaign, consulte "Especificaciones de formato" en la página 9.

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

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(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.
TEMP = VARIANCE(V1:V3, COL)
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.
TEMP = VARIANCE_(V1[1:5]:V3, COL) o TEMP = VARIANCE(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 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.
TEMP = VARIANCE(V1:V3, ROW)
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.
TEMP = VARIANCE(V1[1:5]:V3, ROW) o TEMP = VARIANCE(V1[1:5]:V3[1:5], ROW)
Crea una nueva columna denominada TEMP, donde las 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.

Macro WEEKDAY

La macro WEEKDAY solo está disponible en IBM Campaign.

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 la guía para su producto IBM .

conversion_keyword

Esta palabra clave opcional especifica cómo interpretar formatos de texto para fechas y horas. Seleccione una de las opciones siguientes:

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

TEMP = WEEKDAY("1/1/95")
Crea una nueva columna denominada TEMP que contiene el número 0 (1 de enero de 1995 es un domingo).
TEMP = WEEKDAY(V1, 2)
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???).

TEMP = WEEKDAY(V1:V3, 3) 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 ???).
TEMP = WEEKDAY(V1[10:20]:V2, 10) 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 ???).

Funciones relacionadas

Función	Descripción
NUMBER	Convierte cadenas de texto ASCII para horas y fechas en valores numéricos

Macro WEEKDAYOF

La macro WEEKDAYOF solo está disponible en IBM Campaign.

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 "Macro DATE" en la página 51 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.

Macro XOR

La macro XOR está disponible en IBM Campaign y IBM Interact.

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 cualquiera de éstos. 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 .

`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 cualquiera de éstos. 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 la guía para su producto 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` 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 `data2`, 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 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

Macro YEAROF

La macro YEAROF solo está disponible en IBM Campaign.

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 `date_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 “Macro `DATE`” en la página 51.

Funciones relacionadas

Función	Descripción
<code>DAYOF</code>	Devuelve el día actual del mes en forma de número.
<code>MONTHOF</code>	Devuelve el mes del año como número.
<code>WEEKDAYOF</code>	Devuelve el día de la semana como número.

Before you contact 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 these guidelines 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.

Nota: Technical Support does not write or create API scripts. For assistance in implementing our API offerings, contact IBM Professional Services.

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 that 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, check for a `version.txt` file that is located under the installation directory for your application.

Contact information for IBM technical support

For ways to contact IBM technical support, see the IBM Product Technical Support website: (http://www.ibm.com/support/entry/portal/open_service_request).

Nota: To enter a support request, you must log in with an IBM account. This account must be linked to your IBM customer number. To learn more about associating your account with your IBM customer number, see **Support Resources > Entitled Software Support** on the Support Portal.

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 on 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.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, 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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites 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
B1WA LKG1
550 King Street
Littleton, MA 01460-1250
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 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.

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