Versão 9 Release 1.2
23 de setembro de 2015

Guia do Usuário do IBM Macros for IBM EMM

Observação

Antes de usar estas informações e o produto suportado por elas, leia as informações em “Avisos” na página 161.

Esta edição se aplica à versão 9, liberação 1, modificação 2 da família de produtos do IBMCampaign e a todas as liberações e modificações subsequentes, até que seja indicado de outra forma em novas edições.

© Copyright IBM Corporation 1998, 2015.

Índice

Capítulo 1. Utilizando macros no IBM Campaign 1

Resumos da função macro para o IBM Campaign .. 1
- Funções Estatísticas 1
- Funções Matemáticas e Trigonométricas 2
- Funções de Sequência de Caracteres 6
- Funções de Data e Hora 7
- Funções de Agrupamento 8
- Funções Diversas 8

Parâmetros da função macro para o IBM Campaign . 9
- Especificações de Formato 9
- Uso de Constantes 9

Capítulo 2. Utilizando macros no IBM Interact 11

Resumos da função macro para o IBM Interact .. 11
- Funções Estatísticas 11
- Funções Matemáticas e Trigonométricas 12
- Funções de Sequência de Caracteres 13
- Funções de Data e Hora 14
- Funções Diversas 15

Parâmetros da função macro para o IBM Interact .. 15
- Especificações de Formato 15
- Uso de Constantes 16

Capítulo 3. Referência do IBM Macros 17

Palavras-Chave de Formatos Válidos 17
Macro ABS 19
Macro ACOS 20
Macro ACOT 21
Macro ADD_MONTHS 23
Macro AND 24
Macro ASIN 25
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 37
Macro COLUMN 39
Macro COS 40
Macro COSH 41
Macro COT 43
Macro COUNT 45
Macro CURRENT_DATE 46
Macro CURRENT_DAY 47
Macro CURRENT_JULIAN 47
Macro CURRENT_MONTH 48
Macro CURRENT_TIME 48
- Configuração de data em seu aplicativo da web 49

Macro CURRENT_WEEKDAY 50
Macro CURRENT_YEAR 50
Macro DATE 51
Macro DATE_FORMAT 52
Macro DATE_JULIAN 53
Macro DATE_STRING 54
Macro DAY_BETWEEN 56
Macro DAY_FROMNOW 57
Macro DAY_INTERVAL 57
Macro DAYOF 58
Macro DISTANCE 59
Macro DIV 60
Macro EQ 61
Macro EXP 63
Macro EXTERNALCALLOUT 65
Macro FACTORIAL. 65
Macro FLOOR 66
Macro FORMAT 67
Macro FRAÇÃO 70
Macro GE 72
Macro GROUPBY 73
macro GROUPBY_WHERE 76
Macro GT 77
Macro IF 79
Macro IN 80
Macro INT 81
Macro INVERSE. 83
Macro IS 84
Macro ISERROR 84
Macro ISODD 85
Macro ISEVEN 86
Macro ISODD 87
Macro LE 88
Macro LIKE 90
Macro LN ou LOG 92
Macro LOG2 93
Macro LOG10 94
Macro LOWER 95
Macro LT 96
Macro LTRIM. 98
Macro MAX 98
Macro MEAN 100
Macro MIN 102
Macro MINUS 103
Macro MOD. 105
Macro MONTHOF 107
Macro MULT 107
Macro NE 109
Macro NOT 111
Macro NUMBER 112
Macro OR 118
Macro POSITION 120
Macro PLUS. 122
Macro POW 123
Macro RANDOM 125
Macro RANDOM_GAUSS 126
Macro ROUND. 127
Macro ROWNUM 128
Macro RTRIM 129

© Copyright IBM Corp. 1998, 2015

Macro SIGN 129
Macro SIN 130
Macro SINH 132
Macro SQRT 133
Macro STDV ou STDEV 134
Macro STRING_CONCAT 136
Macro STRING_HEAD 138
Macro STRING_LENGTH 139
Macro STRING_PROPER 140
Macro STRING_SEG 141
Macro STRING_TAIL 142
Macro SUBSTR ou SUBSTRING 143
Macro SUM 144
Macro TAN 146
Macro TANH 147
Macro TOTAL 149
Macro TRUNCATE 150
Macro UPPER 152
Macro VARIANCE 152
Macro WEEKDAY 154
Macro WEEKDAYOF 155
Macro XOR 156
Macro YEAROF 158

Entrando em contato com o suporte técnico do IBM 159

Avisos 161
Marcas Comerciais 163
Política de Privacidade e Termos de Considerações de Uso 163

Capítulo 1. Utilizando macros no IBM Campaign

Este capítulo fornece informações de uso sobre macros do IBM® Campaign. Assegure-se de ler este capítulo antes de tentar utilizar macros no IBM Campaign.

Resumos da função macro para o IBM Campaign

As tabelas nesta seção resumem as funções macro por categoria.

Páginas de referência detalhadas para cada função macro são fornecidas em ordem alfabética em Capítulo 3, "Referência do IBM Macros", na página 17.

Importante: As macros podem se aplicar ao IBM Campaign e ao IBM Interact ou a apenas um desses produtos. As descrições de macro identificam os produtos nos quais elas estão disponíveis.

Consulte "Parâmetros da função macro para o IBM Campaign" na página 9 para obter informações sobre os parâmetros de entrada de função macro.

Informações relacionadas:

"Funções Estatísticas"

"Funções Matemáticas e Trigonométricas" na página 2

"Funções de Sequência de Caracteres" na página 6

"Funções de Data e Hora" na página 7

"Funções de Agrupamento" na página 8

"Funções Diversas" na página 8

Funções Estatísticas

Nome da Macro	Retorna	Descrição
AVG	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula a média aritmética ou a média de um intervalo de células
COUNT	Valor único em uma nova coluna.	Conta o número de valores em um intervalo de dados especificado

© Copyright IBM Corp. 1998, 2015

Nome da Macro	Retorna	Descrição
MAX	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula o máximo de um intervalo de células
MEAN	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula a média aritmética ou a média de um intervalo de células
MIN	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula o mínimo de um intervalo de células
STDV ou STDEV	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula o desvio padrão de um intervalo de células
VARIANCE	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula a variância de um intervalo de células

Funções Matemáticas e Trigonométricas

Nome da Macro	Retorna	Descrição
ABS	Uma coluna para cada coluna de entrada	Calcula o valor absoluto do conteúdo do intervalo de dados especificado
ACOS	Uma coluna para cada coluna de entrada	Calcula o arco-cosseno do conteúdo do intervalo de dados especificado

Nome da Macro	Retorna	Descrição
ACOT	Uma coluna para cada coluna de entrada	Calcula o arco-cotangente do conteúdo do intervalo de dados especificado
ASIN	Uma coluna para cada coluna de entrada	Calcula o arco-seno do conteúdo do intervalo de dados especificado
ATAN	Uma coluna para cada coluna de entrada	Calcula o arco-tangente do conteúdo do intervalo de dados especificado
AVG	Uma coluna para cada coluna de entrada	Calcula a média aritmética ou a média de células no intervalo de dados especificado
BETWEEN	Uma coluna para cada coluna de entrada	Compara dois valores para determinar se o valor fornecido está entre dois outros valores
CEILING	Uma coluna para cada coluna de entrada	Calcula o limite de cada valor no intervalo de dados especificado
COLUMN	Uma coluna para cada coluna de entrada	Cria novas colunas, concatenando verticalmente os valores de entrada em cada coluna
COS	Uma coluna para cada coluna de entrada	Calcula o cosseno do conteúdo do intervalo de dados especificado
COSH	Uma coluna para cada coluna de entrada	Calcula o cosseno hiperbólico do conteúdo do intervalo de dados especificado
COT	Uma coluna para cada coluna de entrada	Calcula a cotangente do conteúdo do intervalo de dados especificado
COUNT	Uma coluna contendo um valor único	Conta o número de células contendo valores no intervalo de dados especificado
EXP	Uma coluna para cada coluna de entrada	Calcula um ou mais números naturais elevados ao conteúdo de cada célula no intervalo de dados especificado
FACTORIAL	Uma coluna para cada coluna de entrada	Calcula o fatorial de cada valor no intervalo de dados especificado
FLOOR	Uma coluna para cada coluna de entrada	Calcula o piso de cada valor no intervalo de dados especificado.
FRACTION	Uma coluna para cada coluna de entrada	Retorna a parte fracionária de cada valor no intervalo de dados especificado.

Nome da Macro	Retorna	Descrição
INT	Uma coluna para cada coluna de entrada	Calcula o valor de número inteiro (arredondado para baixo) do conteúdo do intervalo de dados especificado
INVERSE	Uma coluna para cada coluna de entrada	Calcula o negativo do conteúdo do intervalo de dados especificado
LN	Uma coluna para cada coluna de entrada	Calcula o log natural do conteúdo do intervalo de dados especificado
LOG	Uma coluna para cada coluna de entrada	Calcula o log natural do conteúdo do intervalo de dados especificado
LOG2	Uma coluna para cada coluna de entrada	Calcula o log de base2 do conteúdo do intervalo de dados especificado
LOG10	Uma coluna para cada coluna de entrada	Calcula o log de base10 do conteúdo do intervalo de dados especificado
MAX	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula o máximo de um intervalo de células
MEAN	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula a média aritmética ou a média de um intervalo de células
MIN	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula o mínimo de um intervalo de células
RANDOM	Uma coluna com o número especificado de valores	Retorna o número especificado de números aleatórios
RANDOM_GAUSS	Uma coluna com o número especificado de valores	Retorna o número especificado de valores aleatórios a partir de uma distribuição gaussiana

Nome da Macro	Retorna	Descrição
ROUND	Uma coluna para cada coluna de entrada	Calcula o valor arredondado do conteúdo do intervalo de dados especificado
SIGN	Uma coluna para cada coluna de entrada	Calcula o sinal (positivo ou negativo) dos valores no intervalo de dados especificado
SIN	Uma coluna para cada coluna de entrada	Calcula o seno do conteúdo do intervalo de dados especificado
SINH	Uma coluna para cada coluna de entrada	Calcula o seno hiperbólico do conteúdo do intervalo de dados especificado
SQRT	Uma coluna para cada coluna de entrada	Calcula a raiz quadrada do conteúdo do intervalo de dados especificado
STDV ou STDEV	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula o desvio padrão de um intervalo de células
SUM	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula a soma de um intervalo de células
TAN	Uma coluna para cada coluna de entrada	Calcula a tangente do conteúdo do intervalo de dados especificado
TANH	Uma coluna para cada coluna de entrada	Calcula a tangente hiperbólica do conteúdo do intervalo de dados especificado
TOTAL	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula a soma de um intervalo de células
TRUNCATE	Uma coluna para cada coluna de entrada	Retorna a parte não fracionária de cada valor no intervalo de dados especificado.

Nome da Macro	Retorna	Descrição
VARIANCE	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula a variância de um intervalo de células

Funções de Sequência de Caracteres

Nome da Macro	Retorna	Descrição
FORMAT	Uma coluna para cada coluna de entrada	Fornece controle de formatação de saída para números e sequências (como largura da saída, alinhamento, precisão numérica, símbolo de ponto decimal, símbolo de agrupamento, e assim por diante). Retorna a sequência de saída formatada.
LIKE	Uma coluna para cada coluna de entrada	Determina se uma sequência de texto corresponde a um padrão especificado
LOWER	Uma coluna para cada coluna de entrada	Converte valor de sequência em minúsculas
LTRIM	Uma coluna para cada coluna de entrada	Remove caracteres de espaço à esquerda de cada valor de sequência
NUMBER	Uma coluna para cada coluna de entrada	Converte sequências de texto ASCII de datas e horas em valores numéricos.
POSITION	Uma coluna para cada coluna de entrada	Retorna a posição inicial de um padrão em uma sequência de texto
RTRIM	Uma coluna para cada coluna de entrada	Remove caracteres de espaços à direita de cada valor de sequência
STRING_CONCAT	Uma coluna com um valor para cada linha da coluna de entrada mais curta	Concatena sequências de texto a partir dos intervalos de dados especificados
STRING_HEAD	Uma coluna para cada coluna de entrada	Retorna os primeiros *n* caracteres de cada sequência no intervalo de dados especificado
STRING_LENGTH	Uma coluna para cada coluna de entrada	Retorna o comprimento de cada sequência no intervalo de dados especificado.

Nome da Macro	Retorna	Descrição
STRING_PROPER	Uma coluna para cada coluna de entrada	Converte cada valor de sequência ao alterar a primeira letra ou qualquer letra após um caractere de espaço em branco ou símbolo (diferente de sublinhado) em letras maiúsculas, e todos os outros caracteres em letras minúsculas.
STRING_SEG	Uma coluna para cada coluna de entrada	Retorna o segmento da sequência entre dois índices especificados
STRING_TAIL	Uma coluna para cada coluna de entrada	Retorna os últimos *n* caracteres de cada sequência no intervalo de dados especificado
SUBSTR ou SUBSTRING	Uma coluna para cada coluna de entrada	Retorna os caracteres de uma sequência a partir de uma posição inicial
UPPER	Uma coluna para cada coluna de entrada	Converte valor de sequência em maiúsculas

Funções de Data e Hora

Nome da Macro	Retorna	Descrição
ADD_MONTHS	Uma coluna para cada coluna de entrada	Retorna uma data após incluir o número especificado de meses
CURRENT_DATE	Uma coluna para cada coluna de entrada	Retorna a data atual em `format`
CURRENT_DAY	Uma coluna para cada coluna de entrada	Retorna o dia atual do mês como um número de 1-31
CURRENT_JULIAN	Uma coluna para cada coluna de entrada	Retorna o número Juliano para a data atual
CURRENT_MONTH	Uma coluna para cada coluna de entrada	Retorna o mês atual do ano como um número de 1-12.
CURRENT_TIME	Uma coluna para cada coluna de entrada	Retorna a hora atual como uma sequência
CURRENT_WEEKDAY	Uma coluna para cada coluna de entrada	Retorna o dia da semana atual do mês como um número de 0-6
CURRENT_YEAR	Uma coluna para cada coluna de entrada	Retorna o ano atual como um número
DATE	Uma coluna para cada coluna de entrada	Converte uma sequência de datas em uma data juliana
DATE_FORMAT	Uma coluna para cada coluna de entrada	Transforma formatos de data
DATE_JULIAN	Uma coluna para cada coluna de entrada	Retorna a data juliana

Nome da Macro	Retorna	Descrição
DATE_STRING	Uma coluna para cada coluna de entrada	Retorna a sequência de datas da data Juliana
DAY_BETWEEN	Uma coluna para cada coluna de entrada	Retorna o número de dias entre duas datas
DAY_FROMNOW	Uma coluna para cada coluna de entrada	Retorna o número de dias entre a data atual e a data especificada
DAY_INTERVAL	Uma coluna para cada coluna de entrada	Retorna o número de dias entre duas datas
DAYOF	Uma coluna para cada coluna de entrada	Retorna o dia do mês como um número
MONTHOF	Uma coluna para cada coluna de entrada	Retorna o mês do ano como um número
WEEKDAY	Uma coluna para cada coluna de entrada	Converte sequências de datas de texto ASCII no dia da semana
WEEKDAYOF	Uma coluna para cada coluna de entrada	Retorna o dia da semana como um número
YEAROF	Uma coluna para cada coluna de entrada	Retorna o ano como um número

Funções de Agrupamento

Nome da Macro	Retorna	Descrição
GROUPBY	Uma nova coluna com um valor para cada linha	Resume entre diversas linhas de dados em um grupo
GROUPBY_WHERE	Uma nova coluna com um valor para cada linha	Resume entre diversas linhas de dados que atendem a uma condição especificada e que estão em um grupo

Funções Diversas

Nome da Macro	Retorna	Descrição
IF	Uma coluna com um valor para cada linha da coluna de entrada mais curta	Inicia uma instrução if-then-else condicional
ISERROR	Uma coluna com um valor para cada linha da coluna de entrada mais curta	Retornará um se qualquer valor na linha de entrada contiver uma célula com erro (???), caso contrário, zero
ISEVEN	Uma coluna para cada coluna de entrada	Testa se os valores de entrada são pares (ou seja, divisíveis por dois)
ISODD	Uma coluna para cada coluna de entrada	Testa se os valores de entrada são ímpares (ou seja, não divisíveis por dois)

Nome da Macro	Retorna	Descrição
ROWNUM	Uma coluna para cada coluna de entrada	Gera números sequenciais de um até o número de registros.

Parâmetros da função macro para o IBM Campaign

Esta seção descreve os parâmetros e o uso das funções macro no IBM Campaign.

Especificações de Formato

Esta seção descreve o formato para alguns parâmetros mais normalmente utilizados. Ela se aplica a todas as referências a esses parâmetros pelas especificações de função macro neste capítulo.

dados

O parâmetro `data` representa uma coluna de dados para uma função de macro na qual agir.

Pode ser uma constante ou um campo. Consulte a função macro específica para obter detalhes.

Nota: O IBM Campaign não suporta cálculos em vários campos ao mesmo tempo ou em um subconjunto de linhas.

Alguns outros nomes de parâmetro também utilizam o mesmo formato que `data`. As descrições desses parâmetros referenciam esta seção e formato.

Palavra-chave

O parâmetro `keyword` controla o comportamento da função macro. Ele indica que uma palavra-chave poderá ser especificada (se for omitida, o padrão será utilizado). As opções de palavra-chave são listadas para cada função macro individual no seguinte formato:

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

Selecione a opção de palavra-chave que fornece o comportamento desejado. A opção padrão é mostrada em negrito. Por exemplo, dadas as opções a seguir:

```
{RADIANS | DEGREES}
```

As duas funções macro a seguir são válidas:

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

Nota: Muitas funções macro aceitam os parâmetros de palavra-chave {ALL | COL | ROW}. Estas palavras-chave não se aplicam ao IBM Campaign porque os dados de entrada são sempre uma coluna ou um campo único. A macro sempre se comporta como se a palavra-chave COL fosse especificada. Portanto, não é necessário especificar estas palavras-chave ao usar o IBM Campaign.

Uso de Constantes

A maioria dos parâmetros de função macro pode utilizar constantes numéricas ou expressões que são avaliadas para uma constante numérica (as funções macro que operam em sequências podem utilizar constantes de sequência).

Nas funções macro que executam operações de registro por registro (por exemplo, inclusão de duas colunas numéricas), o uso de uma constante é equivalente a especificar uma coluna que contenha esse valor constante em cada linha. Basicamente, quando uma constante é fornecida como um parâmetro de entrada, a constante é expandida para o mesmo comprimento que a entrada.

Algumas funções macro podem utilizar sequências de texto ASCII e constantes numéricas. Os parâmetros que podem aceitar constantes numéricas e sequências de texto ASCII são indicados na seção "Parâmetros" de cada função macro.

Exemplos são fornecidos na tabela a seguir.

Definição da Função	**Como a Constante É Interpretada**
`PERCENT_UTILIZ = (CURR_BAL*100)/ CREDIT_LIM`	A constante `100` é interpretada como uma coluna contendo o mesmo número de linhas que a coluna `CURR_BAL`, com cada linha contendo a constante `100`. O campo derivado `PERCENT_UTILIZ` conterá cada valor de `CURR_BAL` multiplicado por `100` e dividido por cada valor de `CREDIT_LIM`.
`NAME = STRING_CONCAT ("Mr. ", LAST_NAME)`	A constante `"Mr."` é interpretada como uma coluna contendo o mesmo número de linhas que a coluna `LAST_NAME`, com cada linha contendo a constante `"Sr."` . O campo derivado `NAME` conterá cada uma das sequências de texto em `LAST_NAME` prefixados por `"Mr."` .

Capítulo 2. Utilizando macros no IBM Interact

Este capítulo fornece informações de uso sobre macros do IBM Interact. Assegure-se de ler este capítulo antes de tentar utilizar macros no IBM Interact.

Auxiliar de Fórmula e Verificação de Sintaxe da Expressão da Macro

Importante: A caixa de diálogo Auxiliar de Fórmula, incluindo sua função de Verificação de Sintaxe, valida atualmente as expressões macro de acordo com o que o IBM Campaign suporta. No entanto, o IBM Interact suporta apenas um subconjunto de funcionalidade de macro do Campaign. Portanto, assegure-se de que as macros e as palavras-chave (como palavras-chave de formato de data) usadas para o Interact sejam suportadas. Procure notas relacionadas ao Interact no capítulo Referência de Macros deste guia.

Resumos da função macro para o IBM Interact

As tabelas nas seções a seguir fornecem descrições detalhadas das macros que são específicas para o IBM Interact.

Páginas de referência detalhadas para cada função macro são fornecidas em ordem alfabética em Capítulo 3, “Referência do IBM Macros”, na página 17.

Importante: As macros podem se aplicar ao IBM Campaign e ao IBM Interact ou a apenas um desses produtos. As descrições de macro identificam os produtos nos quais elas estão disponíveis.

O “Parâmetros da função macro para o IBM Interact” na página 15 fornece informações sobre os parâmetros de entrada da função macro para o Interact.

Informações relacionadas:

“Funções Estatísticas”

“Funções Matemáticas e Trigonométricas” na página 12

“Funções de Sequência de Caracteres” na página 13

“Funções de Data e Hora” na página 14

“Funções Diversas” na página 15

Funções Estatísticas

Nome da Macro	Retorna	Descrição
AVG	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula a média aritmética ou a média de um intervalo de células

© Copyright IBM Corp. 1998, 2015

Nome da Macro	Retorna	Descrição
MAX	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula o máximo de um intervalo de células
MEAN	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula a média aritmética ou a média de um intervalo de células
MIN	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula o mínimo de um intervalo de células
STDV ou STDEV	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula o desvio padrão de um intervalo de células

Funções Matemáticas e Trigonométricas

Nome da Macro	Retorna	Descrição
AVG	Uma coluna para cada coluna de entrada	Calcula a média aritmética ou a média de células no intervalo de dados especificado
MAX	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula o máximo de um intervalo de células

Nome da Macro	Retorna	Descrição
MEAN	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula a média aritmética ou a média de um intervalo de células
MIN	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula o mínimo de um intervalo de células
STDV ou STDEV	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula o desvio padrão de um intervalo de células
SUM	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula a soma de um intervalo de células
TOTAL	Valor único em uma nova coluna para a palavra-chave ALL; uma coluna com um valor único para cada coluna de entrada para a palavra-chave COL; uma coluna com um valor para cada linha para a palavra-chave ROW.	Calcula a soma de um intervalo de células

Funções de Sequência de Caracteres

Nome da Macro	Retorna	Descrição
LIKE	Uma coluna para cada coluna de entrada	Determina se uma sequência de texto corresponde a um padrão especificado
LOWER	Uma coluna para cada coluna de entrada	Converte valor de sequência em minúsculas

Nome da Macro	Retorna	Descrição
LTRIM	Uma coluna para cada coluna de entrada	Remove caracteres de espaço à esquerda de cada valor de sequência
NUMBER	Uma coluna para cada coluna de entrada	Converte sequências de texto ASCII de datas e horas em valores numéricos.
RTRIM	Uma coluna para cada coluna de entrada	Remove caracteres de espaços à direita de cada valor de sequência
STRING_CONCAT	Uma coluna com um valor para cada linha da coluna de entrada mais curta	Concatena sequências a partir dos intervalos de dados especificados
SUBSTR ou SUBSTRING	Uma coluna para cada coluna de entrada	Retorna os caracteres de uma sequência a partir de uma posição inicial
UPPER	Uma coluna para cada coluna de entrada	Converte valor de sequência em maiúsculas

Funções de Data e Hora

Nome da Macro	Retorna	Descrição
ADD_MONTHS	Uma coluna para cada coluna de entrada	Inclui o número especificado de meses e retorna uma data
CURRENT_DATE	Uma coluna para cada coluna de entrada	Retorna a data atual em `format`
CURRENT_DAY	Uma coluna para cada coluna de entrada	Retorna o dia atual do mês como um número no intervalo de 1-31
CURRENT_MONTH	Uma coluna para cada coluna de entrada	Retorna o mês atual do ano como um número no intervalo de 1-12
CURRENT_WEEKDAY	Uma coluna para cada coluna de entrada	Retorna o dia da semana atual do mês como um número no intervalo de 0-6
CURRENT_YEAR	Uma coluna para cada coluna de entrada	Retorna o ano atual como um número
DATE	Uma coluna para cada coluna de entrada	Converte uma sequência de datas em uma data juliana
DATE_FORMAT	Uma coluna para cada coluna de entrada	Transforma formatos de data

Funções Diversas

Nome da Macro	Retorna	Descrição
EXTERNALCALLOUT	Valores conforme definido pelo aplicativo customizado gravado com a API ExternalCallout	Chama um aplicativo customizado gravado com a API ExternalCallout. Para obter mais informações, consulte o *Guia do Desenvolvedor do IBM Interact*.
IF	Uma coluna com um valor para cada linha da coluna de entrada mais curta	Inicia uma instrução if-then-else condicional

Parâmetros da função macro para o IBM Interact

Esta seção descreve os parâmetros e o uso das funções macro no IBM Interact.

Especificações de Formato

Esta seção descreve o formato para alguns parâmetros mais normalmente utilizados. Ela se aplica a todas as referências a esses parâmetros pelas especificações de função macro nesta seção.

dados

O parâmetro `data` representa uma coluna de dados para uma função de macro na qual agir. Pode ser uma constante ou um campo. Consulte a função macro específica para obter detalhes.

Nota: O IBM Interact não suporta cálculos em vários campos ao mesmo tempo ou em um subconjunto de linhas.

Alguns outros nomes de parâmetro também utilizam o mesmo formato que `data`. A descrição desses parâmetros referencia esta seção e formato.

Palavra-chave

O parâmetro `keyword` controla o comportamento da função macro. Ele indica que uma palavra-chave poderá ser especificada (se for omitida, o padrão será utilizado). As opções de palavra-chave são listadas para cada função macro individual no seguinte formato:

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

Selecione a opção de palavra-chave que fornece o comportamento desejado. A opção padrão é mostrada em negrito. Por exemplo, dadas as opções a seguir:

```
{RADIANS | DEGREES}
```

As duas funções macro a seguir são válidas:

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

Nota: Muitas funções de macro utilizam os parâmetros de palavra-chave `{ALL | COL | ROW}`. Estas palavras-chave não se aplicam no IBM Interact porque os dados de entrada são sempre uma coluna ou um campo único. A macro sempre se

comporta como se a palavra-chave COL fosse especificada. Portanto, não é necessário especificar estas palavras-chave ao usar o IBM Interact.

Uso de Constantes

A maioria dos parâmetros de função macro pode utilizar constantes numéricas ou expressões que são avaliadas para uma constante numérica (as funções macro que operam em sequências podem utilizar constantes de sequência).

Nas funções macro que executam operações de registro por registro (por exemplo, inclusão de duas colunas numéricas), o uso de uma constante é equivalente a especificar uma coluna que contenha esse valor constante em cada linha. Basicamente, quando uma constante é fornecida como um parâmetro de entrada, a constante é expandida para o mesmo comprimento que a entrada.

Algumas funções macro podem utilizar sequências de texto ASCII e também constantes numéricas. Os parâmetros que podem aceitar constantes numéricas e sequências de texto ASCII são indicados na seção "Parâmetros" de cada função macro.

Exemplos são fornecidos na tabela a seguir.

Definição da Função	**Como a Constante É Interpretada**
`PERCENT_UTILIZ = (CURR_BAL*100)/ CREDIT_LIM`	A constante `100` é interpretada como uma coluna contendo o mesmo número de linhas que a coluna `CURR_BAL`, com cada linha contendo a constante `100`. O campo derivado `PERCENT_UTILIZ` conterá cada valor de `CURR_BAL` multiplicado por `100` e dividido por cada valor de `CREDIT_LIM`.
`NAME = STRING_CONCAT ("Mr.", LAST_NAME)`	A constante `"Mr."` é interpretada como uma coluna contendo o mesmo número de linhas que a coluna `LAST_NAME`, com cada linha contendo a constante `"Sr."` . O campo derivado `NAME` conterá cada uma das sequências de texto em `LAST_NAME` prefixados por `"Mr."` .

Nota: Constantes, como DT_DELIM_M_D_A, requerem aspas simples.

Capítulo 3. Referência do IBM Macros

Esta seção descreve cada macro disponível que está disponível para uso no IBM Campaign, IBM Interact, ou em ambos. As macros são listadas em ordem alfabética.

Importante: Não utilize nomes de função ou palavras-chave da Idioma da Macro do IBM para títulos de coluna nas tabelas de usuário no IBM Campaign, não importa se estiver mapeando a partir de um banco de dados ou de um arquivo simples. Essas palavras reservadas podem causar erros, se usadas nos títulos de coluna em tabelas mapeadas.

Palavras-Chave de Formatos Válidos

A tabela a seguir mostra as palavras-chave para os formatos válidos, com uma descrição e exemplo de cada uma.

Palavra-chave	Descrição	Exemplo(s)
MM	mês de 2 dígitos	01, 02, 03, ..., 12
MMDD	mês de 2 dígitos e dia de 2 dígitos	31 de março é 0331
MMDDYY	mês de 2 dígitos, dia de 2 dígitos e ano de 2 dígitos	31 de março de 1970 é 033170
MMDDYYYY	mês de 2 dígitos, dia de 2 dígitos e ano de 4 dígitos	31 de março de 1970 é 03311970
DELIM_M_D	Qualquer mês delimitado seguido por dia	31 de março, 3/31 ou 03-31
DELIM_M_D_Y	Qualquer mês, dia e ano delimitado	31 de março de 1970 ou 3/31/70
DELIM_Y_M	Qualquer ano delimitado seguido por mês	Março, 70; 3-70 ou 3/1970
DELIM_Y_M_D	Qualquer ano, mês e dia delimitados	31 de março de 1970 ou 70/3/31
YYMMM	Ano de 2 dígitos e mês de 3 letras	70MAR
YYMMMDD	Ano de 2 dígitos, mês de 3 letras e dia de 2 dígitos	70MAR31
YY	Ano de 2 dígitos	70
YYMM	Ano de 2 dígitos e mês de 2 dígitos	7003
YYMMDD	Ano de 2 dígitos, mês de 2 dígitos e dia de 2 dígitos	700331
YYYYMMM	Ano de 4 dígitos e mês de 3 letras	1970MAR
YYYYMMMDD	Ano de 4 dígitos, mês de 3 letras e dia de 2 dígitos	1970MAR31
YYYY	Ano de 4 dígitos	1970

© Copyright IBM Corp. 1998, 2015

Palavra-chave	Descrição	Exemplo(s)
YYYYMM	Ano de 4 dígitos e mês de 2 dígitos	197003
YYYYMMDD	Ano de 4 dígitos, mês de 2 dígitos e dia de 2 dígitos	19700331
DELIM_M_Y	Qualquer mês delimitado seguido pelo ano	3-70, 3/70, mar 70, março 1970
DELIM_D_M	Qualquer dia delimitado seguido pelo mês	31-3, 31/3, 31 março
DELIM_D_M_Y	Qualquer dia, mês e ano delimitados	31-MAR-70, 31/3/1970, 31 03 70
DD	Dia de 2 dígitos	31
DDMMM	Dia de 2 dígitos e mês de 3 letras	31MAR
DDMMMYY	Dia de 2 dígitos, mês de 3 letras e ano de 2 dígitos	31MAR70
DDMMMYYYY	Dia de 2 dígitos, mês de 3 letras e ano de 4 dígitos	31MAR1970
DDMM	Dia de 2 dígitos e mês de 2 dígitos	3103
DDMMYY	Dia de 2 dígitos, mês de 2 dígitos e ano de 2 dígitos	310370
DDMMYYYY	Dia de 2 dígitos, mês de 2 dígitos e ano de 4 dígitos	31031970
MMYY	Mês de 2 dígitos e ano de 2 dígitos	0370
MMYYYY	Mês de 2 dígitos e ano de 4 dígitos	031970
MMM	Mês de 3 letras	MAR
MMMDD	Mês de 3 letras e dia de 2 dígitos	MAR31
MMMDDYY	Mês de 3 letras, dia de 2 dígitos e ano de 2 dígitos	MAR3170
MMMDDYYYY	Mês de 3 letras, dia de 2 dígitos e ano de 4 dígitos	MAR311970
MMMYY	Mês de 3 letras e ano de 2 dígitos	MAR70
MMMYYYY	Mês de 3 letras e ano de 4 dígitos	MAR1970
MONTH	Mês do ano	Janeiro, Fevereiro, Março, e assim por diante, ou Jan, Fev, Mar, e assim por diante
WEEKDAY	Dia da semana	Domingo, Segunda-feira, Terça-feira, e assim por diante (Domingo = 0)
WKD	Dia da semana abreviado	Dom, Seg, Ter e assim por diante (Dom = 0)

Macro ABS

A macro ABS está disponível apenas no IBM Campaign.

Sintaxe

```
ABS(data)
```

Parâmetros

`data`

Os valores numéricos para os quais calcular o valor absoluto. Esse parâmetro pode ser um valor constante, uma coluna, um intervalo de células, ou uma expressão que é avaliada para qualquer um desses tipos. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

ABS calcula o valor absoluto dos números no intervalo de dados especificado. O valor absoluto de um número é o seu valor sem o seu sinal (ou seja, números positivos são inalterados; números negativos são retornados como números positivos). ABS retorna uma nova coluna para cada coluna de entrada, cada uma contendo o valor absoluto dos números na coluna de entrada correspondente.

Exemplos

`TEMP = ABS(-3)` ou `TEMP = ABS(3)` Cria uma nova coluna denominada `TEMP` contendo o valor 3.
`TEMP = ABS(V1)` Cria uma coluna denominada `TEMP`, em que cada valor é o valor absoluto do conteúdo da coluna `V1`.
`TEMP = ABS(V1:V3)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os valores absolutos do conteúdo da coluna `V1`, os valores da coluna `VX` são os valores absolutos do conteúdo da coluna `V2`, e os valores da coluna `VY` são os valores absolutos do conteúdo da coluna `V3`.
`TEMP = ABS(V1[10:20])` Cria uma coluna denominada `TEMP`, em que as 11 primeiras células contêm os valores absolutos dos valores nas linhas 10-20 da coluna `V1`. Outras células em `TEMP` estão vazias.
`TEMP = ABS(V1[1:5]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são os valores absolutos das linhas correspondentes da coluna `V1`, e os valores na coluna `VX` são os valores absolutos das linhas correspondentes da coluna `V2`.

Funções relacionadas

Função	Descrição
`SIGN`	Calcula o sinal (positivo ou negativo) dos valores no intervalo de dados especificado.

Macro ACOS

O ACOS está disponível apenas no IBM Campaign.

Sintaxe

ACOS(data [, units_keyword])

Parâmetros

dados

Os valores numéricos dos quais calcular o valor do arco-cosseno. Esse parâmetro pode ser um valor constante, uma coluna, um intervalo de células, ou uma expressão que é avaliada para qualquer um desses tipos. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

units_keyword

Essa palavra-chave opcional determina se os valores de entrada e os resultados são interpretados como graus ou radianos. Selecione um dos valores a seguir:

RADIAN - Executa os cálculos em radianos (padrão)

DEGREE - Executa o cálculo em graus

Se esse parâmetro não for especificado, o padrão será radianos. (Para converter de radianos em graus, divida pelo PI e multiplique por 180).

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte “Especificações de Formato” na página 9.

Descrição

ACOS calcula o arco-cosseno dos valores no intervalo de dados especificado. O arco-cosseno é um ângulo cujo cosseno é o conteúdo de cada célula. ACOS retorna uma nova coluna para cada coluna de entrada, cada uma contendo o arco-cosseno dos números na coluna de entrada correspondente.

Se a palavra-chave RADIAN for utilizada, ACOS retornará valores no intervalo de 0 a π. Se a palavra-chave DEGREE for utilizada, ACOS retornará valores no intervalo de 0 – 180.

Nota: O conteúdo da célula de cada coluna especificada deve ter valores entre -1,0 e 1,0 inclusive. Caso contrário, uma célula em branco será retornada para cada entrada inválida.

Exemplos

```
TEMP = ACOS(0) ou TEMP = ACOS(0, 0) ou TEMP = ACOS(0, RADIAN)
```

Cria uma coluna denominada TEMP contendo o valor 1.571 ($\pi / 2$ radianos).

`TEMP = ACOS(0, 1)` ou `TEMP = ACOS(0, DEGREE)` Cria uma coluna denominada `TEMP` contendo o valor `90` (graus).
`TEMP = ACOS(V1)` Cria uma coluna denominada `TEMP`, em que cada valor é o arco-cosseno (em radianos) do conteúdo da coluna `V1`.
`TEMP = ACOS(V1:V3, 1)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os arcos-cossenos do conteúdo da coluna `V1`, os valores da coluna `VX` são os arcos-cossenos do conteúdo da coluna `V2`, e os valores da coluna `VY` são os arcos-cossenos do conteúdo da coluna `V3`. Todos os valores estão em graus.
`TEMP = ACOS(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os arcos-cossenos dos valores nas linhas 10-20 da coluna `V1` (em radianos). Outras células em `TEMP` estão vazias.
`TEMP = ACOS(V1[1:5]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são os arcos-cossenos das linhas correspondentes da coluna `V1`, e os valores na coluna `VX` são os arcos-cossenos das linhas correspondentes da coluna `V2`. Todos os valores estão em radianos.

Funções relacionadas

Função	Descrição
`ACOT`	Calcula o arco-cotangente do conteúdo do intervalo de dados especificado
`ASIN`	Calcula o arco-seno do conteúdo do intervalo de dados especificado
`ATAN`	Calcula o arco-tangente do conteúdo do intervalo de dados especificado
`COS`	Calcula o cosseno do conteúdo do intervalo de dados especificado

Macro ACOT

O `ACOT` está disponível apenas no IBM Campaign.

Sintaxe

```
ACOT(data [, units_keyword])
```

Parâmetros

`data`

Os valores numéricos dos quais calcular o arco cotangente. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`units_keyword`

Essa palavra-chave opcional determina se os valores de entrada e os resultados são interpretados como graus ou radianos. Selecione uma das opções a seguir:

`RADIAN` - Executa os cálculos em radianos (padrão)

`DEGREE` - Executa o cálculo em graus

Se esse parâmetro não for especificado, o padrão será radianos. (Para converter de radianos em graus, divida pelo `PI` e multiplique por 180).

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte "Especificações de Formato" na página 9.

Descrição

`ACOT` retorna o ângulo cuja cotangente é o conteúdo de cada célula. `ACOT` retorna uma nova coluna para cada coluna de entrada, cada uma contendo o arco tangente dos números na coluna de entrada correspondente. Números de ponto flutuante de 64 bits são utilizados.

Exemplos

`TEMP = ACOT(0.5)` ou `TEMP = ACOT(0.5, 0)` ou `TEMP = ACOT(0.5, RADIAN)` Cria uma coluna denominada `TEMP` contendo o valor `2.157` (radianos).
`TEMP = ACOT(1, 1)` ou `TEMP = ACOT(1, DEGREE)` Cria uma coluna denominada `TEMP` contendo o valor `0.022` (1/45) graus.
`TEMP = ACOT(0)` Cria uma coluna denominada `TEMP` contendo o valor `MAX32_Float` em radianos.
`TEMP = ACOT(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é o arco tangente (em radianos) do conteúdo da coluna `V1`.
`TEMP = ACOT(V1:V3, 1)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os arcos tangentes do conteúdo da coluna `V1`, os valores da coluna `VX` são os arcos tangentes do conteúdo da coluna `V2`, e os valores da coluna `VY` são os arcos tangentes do conteúdo da coluna `V3`. Todos os valores estão em graus.
`TEMP = ACOT(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os arcos tangentes dos valores nas linhas 10-20 da coluna `V1` (em radianos). Outras células em `TEMP` estão vazias.
`TEMP = ACOT(V1[1:5]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são os arcos tangentes das linhas correspondentes da coluna `V1`, e os valores na coluna `VX` são os arcos tangentes das linhas correspondentes da coluna `V2`. Todos os valores estão em radianos.

Funções relacionadas

Função	Descrição
ACOS	Calcula o arco-cosseno do conteúdo do intervalo de dados especificado
ASIN	Calcula o arco-seno do conteúdo do intervalo de dados especificado
ATAN	Calcula o arco-tangente do conteúdo do intervalo de dados especificado
COT	Calcula a cotangente do conteúdo do intervalo de dados especificado

Macro ADD_MONTHS

A macro ADD_MONTHS está disponível em IBM Campaign

Sintaxe

```
ADD_MONTHS(months, date_string [, input_format])
```

Parâmetros

`months`

Um número inteiro que representa um número de meses a ser incluído em date_string.

`date_string`

Uma sequência de texto que representa uma data válida no formato DELIM_M_D_Y ou no formato especificado pelo argumento input_format opcional.

`input_format`

O formato que será utilizado para a data calculada. Para obter uma lista de formatos de data suportados, consulte a função DATE_FORMAT. Observe que o `input_format` determina o formato da sequência de entrada e também o formato da sequência de saída.

Descrição

ADD_MONTHS retorna uma data após incluir o número especificado de meses no date_string especificado. A data será retornada no formato padrão (DELIM_M_D_Y) ou no formato especificado pelo argumento input_format opcional. Se desejar um formato diferente como a saída, utilize DATE_FORMAT.

Se aumentar o mês pelo número especificado de meses produzir uma data inválida, então o resultado será calculado como sendo o último dia do mês, conforme mostrado no último exemplo abaixo. Quando necessário, anos bissextos são levados em conta. Por exemplo, incluir um mês para 31-Jan-2012 resultará em 29-Fev-2012.

Exemplos

`ADD_MONTHS(12, '06-25-11')` inclui um ano (12 meses) até a data especificada e retorna a data 25/06/2012.

`ADD_MONTHS(3, '2011-06-25', DT_DELIM_Y_M_D)` inclui três meses na data especificada e retorna a data 25/09/2011.

`ADD_MONTHS(1, '02-28-2011')` retorna a data 28/03/2011.

`ADD_MONTHS(1, '03-31-2012')` retorna a data 30/04/2012.

Funções relacionadas

Função	Descrição
`DATE`	Converte uma sequência de datas em uma data juliana.
`DATE_FORMAT`	Transforma uma data de `input_format` em `output_format`.

Macro AND

A macro `AND` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

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

Parâmetros

`data1`

Os números para AND lógico com os valores em `data2`. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`data2`

Um ou mais números para AND lógico com os valores em `data1`. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em `data2` deve ser igual ao número de colunas em `data1`, a menos que `data2` seja uma constante. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`AND` calcula o AND lógico entre os dois intervalos de dados especificados. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em `data1` incluída logicamente com a coluna correspondente de `data2` (ou seja, a primeira coluna de `data1` é incluída logicamente na primeira coluna de `data`, a segunda coluna com a segunda coluna, e assim por diante).

Se data2 for uma constante, cada valor em data1 será incluído logicamente por este valor. Se data2 contiver uma ou mais colunas, os cálculos serão executados em uma base de linha por linha entre uma coluna data2 e uma coluna de data2. A primeira linha de data1 é incluída logicamente no valor da primeira linha de data2, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: O operador AND pode ser abreviado com um circunflexo (&&). Use e comercial duplo para separar os dois argumentos (por exemplo, para especificar V1 AND 3, é possível simplesmente digitar V1&&3).

Exemplos

TEMP = 1 AND 8 ou TEMP = 1 && 8 Cria uma nova coluna denominada TEMP que contém o valor um (qualquer número diferente de zero será tratado como um).
TEMP = V1 && 1 Cria uma nova coluna denominada TEMP com o valor um para cada valor da coluna V1.
TEMP = V1 && V1 Cria uma nova coluna denominada TEMP com o valor um para cada valor diferente de zero na coluna V1 e o valor zero para cada zero na coluna V1.
TEMP = V1 && V2 Cria uma nova coluna denominada TEMP, em que cada valor é o valor da linha da coluna V1 incluído logicamente com o valor da linha correspondente da coluna V2.
TEMP = V1:V3 && V4:V6 Cria três novas colunas denominadas TEMP, VX e VY. A coluna TEMP contém os valores em V1 incluídos logicamente com os valores da linha correspondente da coluna V4. A coluna VX contém os valores incluídos logicamente das colunas V2 e V5. A coluna VY contém os valores incluídos logicamente das colunas V3 e V6.
TEMP = V1[10:20] && V2 ou TEMP = V1[10:20] && V2[1:11] Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm o resultado de incluir logicamente os valores nas linhas 10-20 da coluna V1 com os valores nas linhas 1-11 da coluna V2. Outras células em TEMP estão vazias.

Funções relacionadas

Função	Descrição
NOT	Calcula o NOT lógico do conteúdo do intervalo de dados especificado
OR	Calcula o OR lógico entre dois intervalos de dados especificados

Macro ASIN

A está disponível apenas no IBM Campaign. ASIN

Sintaxe

```
ASIN(data [, units_keyword])
```

Parâmetros

`data`

Os valores numéricos dos quais calcular o arco seno. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`units_keyword`

Essa palavra-chave opcional determina se os valores de entrada e os resultados são interpretados como graus ou radianos. Selecione uma das opções a seguir:

`RADIAN` - Executa os cálculos em radianos (padrão)

`DEGREE` - Executa o cálculo em graus

Se esse parâmetro não for especificado, o padrão será radianos. (Para converter de radianos em graus, divida pelo `PI` e multiplique por 180).

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte “Especificações de Formato” na página 9.

Descrição

`ASIN` calcula o arco seno dos valores no intervalo de dados especificado. O arco seno é o ângulo cujo seno é o conteúdo de cada célula. `ASIN` retorna uma nova coluna para cada coluna de entrada, cada uma contendo o arco seno dos números na coluna de entrada correspondente.

Se a palavra-chave `RADIAN` for usada, `ASIN` retornará valores no intervalo de $-\pi/2$ a $\pi/2$. Se a palavra-chave `DEGREE` for utilizada, `ASIN` retornará valores no intervalo de –90 a 90.

Nota: O conteúdo da célula de cada coluna especificada deve ter valores entre -1,0 e 1,0 inclusive. Caso contrário, `???` será retornado para cada entrada inválida.

Exemplos

`TEMP = ASIN(0.5)` ou `TEMP = ASIN(0.5, 0)` ou `TEMP = ASIN(0.5, RADIAN)` Cria uma nova coluna denominada `TEMP` contendo o valor `0.524` ($\pi/6$ radianos).
`TEMP = ASIN(0.5, 1)` ou `TEMP = ASIN(0.5, DEGREE)` Cria uma nova coluna denominada `TEMP` contendo o valor `30` (graus).
`TEMP = ASIN(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é o arco seno (em radianos) do conteúdo da coluna `V1`.

`TEMP = ASIN(V1:V3, 1)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os arcos senos do conteúdo da coluna `V1`, os valores da coluna `VX` são os arcos senos do conteúdo da coluna `V2`, e os valores da coluna `VY` são os arcos senos do conteúdo da coluna `V3`. Todos os valores estão em graus.
`TEMP = ASIN(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os arcos senos dos valores nas linhas 10-20 da coluna `V1` (em radianos). Outras células em `TEMP` estão vazias.
`TEMP = ASIN(V1[1:5]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são os arcos senos das linhas correspondentes da coluna `V1`, e os valores na coluna `VX` são os arcos senos das linhas correspondentes da coluna `V2`. Todos os valores estão em radianos.

Funções relacionadas

Função	Descrição
`ACOS`	Calcula o arco-cosseno do conteúdo do intervalo de dados especificado
`ACOT`	Calcula o arco-cotangente do conteúdo do intervalo de dados especificado
`ATAN`	Calcula o arco tangente do conteúdo do intervalo de dados especificado
`SIN`	Calcula o seno do conteúdo do intervalo de dados especificado

Macro ATAN

O `ATAN` está disponível apenas no IBM Campaign.

Sintaxe

```
ATAN(data [, units_keyword])
```

Parâmetros

`data`

Os valores numéricos dos quais calcular o arco tangente. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`units_keyword`

Essa palavra-chave opcional determina se os valores de entrada e os resultados são interpretados como graus ou radianos. Selecione uma das opções a seguir:

`RADIAN` - Executa os cálculos em radianos (padrão)

DEGREE - Executa o cálculo em graus

Se esse parâmetro não for especificado, o padrão será radianos. (Para converter de radianos em graus, divida pelo PI e multiplique por 180).

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte "Especificações de Formato" na página 9.

Descrição

ATAN calcula o arco tangente dos valores no intervalo de dados especificado. O arco tangente é o ângulo cuja tangente é o conteúdo de cada célula. ATAN retorna uma nova coluna para cada coluna de entrada, cada uma contendo o arco tangente dos números na coluna de entrada correspondente.

Se a palavra-chave RADIAN for usada, ATAN retornará valores no intervalo de $-\pi/2$ a $\pi/2$. Se a palavra-chave DEGREE for utilizada, ATAN retornará valores no intervalo de –90 a 90.

Exemplos

`TEMP = ATAN(1)` ou `TEMP = ATAN(1, 0)` ou `TEMP = ATAN(1, RADIAN)` Cria uma nova coluna denominada TEMP contendo o valor 0.785 ($\pi/4$ radianos).
`TEMP = ATAN(1, 1)` ou `TEMP = ATAN(1, DEGREE)` Cria uma nova coluna denominada TEMP contendo o valor 45 (graus).
`TEMP = ATAN(V1)` Cria uma nova coluna denominada TEMP, em que cada valor é o arco tangente (em radianos) do conteúdo da coluna V1.
`TEMP = ATAN(V1:V3, 1)` Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os arcos tangentes do conteúdo da coluna V1, os valores da coluna VX são os arcos tangentes do conteúdo da coluna V2, e os valores da coluna VY são os arcos tangentes do conteúdo da coluna V3. Todos os valores estão em graus.
`TEMP = ATAN(V1[10:20])` Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm os arcos tangentes dos valores nas linhas 10-20 da coluna V1 (em radianos). Outras células em TEMP estão vazias.
`TEMP = ATAN(V1[1:5]:V2)` Cria duas novas colunas denominadas TEMP e VX, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna TEMP são os arcos tangentes das linhas correspondentes da coluna V1, e os valores na coluna VX são os arcos tangentes das linhas correspondentes da coluna V2. Todos os valores estão em radianos.

Funções relacionadas

Função	Descrição
ACOS	Calcula o arco cosseno do conteúdo do intervalo de dados especificado

Função	Descrição
ASIN	Calcula o arco seno do conteúdo do intervalo de dados especificado
ATAN	Calcula o arco tangente do conteúdo do intervalo de dados especificado
TAN	Calcula a tangente do conteúdo do intervalo de dados especificado

Macro AVG

O AVG está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
AVG(data [, keyword])
```

Parâmetros

data

Os valores numéricos para os quais calcular a média aritmética. Esses valores podem ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

keyword

Essa palavra-chave opcional determina como o cálculo será executado sobre o intervalo de dados de entrada. Selecione uma das palavras-chave a seguir:

ALL - Executa o cálculo em todas as células em data (padrão)

COL - Executa o cálculo separadamente para cada coluna de data

ROW - Executa o cálculo separadamente para cada linha de data

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte “Especificações de Formato” na página 9.

Nota: Muitas funções de macro utilizam os parâmetros de palavra-chave {ALL | COL | ROW}. Estas palavras-chave não se aplicam ao **IBM Campaign** porque os dados de entrada são sempre uma coluna ou um campo único. A macro sempre se comporta como se a palavra-chave COL fosse especificada. Portanto, não é necessário especificar estas palavras-chave ao usar o **IBM Campaign**.

Descrição

AVG calcula a média aritmética ou a média das células no intervalo de dados especificado. A média aritmética é calculada ao somar o conteúdo de todas as células e, em seguida, dividir o resultado pelo número de células. O número de colunas que são retornadas por AVG depende da keyword.

- Se keyword for ALL, AVG retornará uma nova coluna, contendo um valor único (a média de todas as células em data).

- Se keyword for COL, AVG retornará uma nova coluna para cada coluna de entrada. Cada nova coluna contém um valor (a média de todas as células na coluna de entrada correspondente).
- Se keyword for ROW, AVG retornará uma nova coluna que contém a média de cada linha de data.

Nota: Células em branco são ignoradas no cálculo.

Nota: AVG é o mesmo que a função macro MEAN.

Exemplos

`TEMP = AVG(V1)` Cria uma coluna denominada TEMP contendo um valor único que é a média aritmética do conteúdo da coluna V1.
`TEMP = AVG(V1:V3)` Cria uma coluna denominada TEMP contendo um valor único que é a média aritmética do conteúdo das colunas V1, V2 e V3.
`TEMP = AVG(V1[10:20])` Cria uma coluna denominada TEMP contendo um valor único que é a média aritmética das células nas linhas 10-20 da coluna V1.
`TEMP = AVG(V1[1:5]:V4)` Cria uma coluna denominada TEMP contendo um valor único que é a média aritmética das células nas linhas 1-5 das colunas V1 a V4.
`TEMP = AVG(V1:V3, COL)` Cria três novas colunas denominadas TEMP, VX e VY. O valor único na coluna TEMP é a média aritmética do conteúdo da coluna V1. O valor único na coluna VX é a média aritmética do conteúdo da coluna V2. O valor único na coluna VY é a média aritmética do conteúdo da coluna V3.
`TEMP = AVG(V1[1:5]:V3, COL)` Cria três novas colunas denominadas TEMP, VX, e VY, cada uma contendo um valor único. O valor na coluna TEMP é a média aritmética das células nas linhas 1-5 da coluna V1. O valor na coluna VX é a média aritmética das células nas linhas 1-5 da coluna V2. O valor na coluna VY é a média aritmética das células nas linhas 1-5 da coluna V3.
`TEMP = AVG(V1, ROW)` Cria uma coluna denominada TEMP, contendo os mesmos valores da coluna V1 (a média aritmética de qualquer número é ele mesmo).
`TEMP = AVG(V1:V3, ROW)` Cria uma coluna denominada TEMP, em que cada entrada na célula é a média aritmética da linha correspondente nas colunas V1, V2 e V3.
`TEMP = AVG(V1[1:5]:V3, ROW)` Cria uma coluna denominada TEMP, em que as células nas linhas 1-5 contêm a média aritmética da linha correspondente nas colunas V1 a V3. Outras células em TEMP estão vazias.

Funções relacionadas

Função	Descrição
SUM ou TOTAL	Calcula a soma de um intervalo de células

Macro BETWEEN

O BETWEEN está disponível apenas no IBM Campaign.

Sintaxe

```
value1 BETWEEN value2 AND value3
```

Parâmetros

```
Equivalent to value1 >= value2 AND < value3
```

Descrição

BETWEEN é uma variante especial do predicado de comparação. Os detalhes deste predicado são importantes e a ordem dos operandos possui algumas implicações inesperadas. Veja a seção de exemplos.

Nota: FROM e FOR utilizam sintaxe idêntica.

Exemplos

```
10 BETWEEN 5 AND 15 É verdadeiro, mas: 10 BETWEEN 15 AND 5 É falso:
```

porque a maneira equivalente de expressar BETWEEN (utilizando AND) possui uma ordem específica que não importa quando você está usando literais, mas poderá importar grandiosamente se você fornecer value2 e value3 usando variáveis do host, parâmetros ou até mesmo subconsultas.

Macro BIT_AND

O BIT_AND está disponível no IBM Campaign e no IBM Interact.

Sintaxe

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

Parâmetros

data1

Os números inteiros não-negativos para executar AND bit a bit com os valores em data2. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

data2

Um ou mais números inteiros não negativos para executar AND bit a bit com os valores em data1. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em data2 deve ser igual ao número de colunas em data1, a menos que data2 seja uma constante. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

BIT_AND executa um AND bit a bit entre dois intervalos de dados especificados. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em data1 com AND bit a bit executado com a coluna correspondente de data2 (ou seja, a primeira coluna de data1 tem o AND bit a bit executado com a primeira coluna de data, a segunda coluna com a segunda coluna, e assim por diante).

Se data2 for uma constante, cada valor em data1 terá o AND bit a bit executado por esse valor. Se data2 contiver uma ou mais colunas, os cálculos serão executados em uma base de linha por linha entre uma coluna data2 e uma coluna de data2. A primeira linha de data1 tem o AND bit a bit executado para o valor da primeira linha de data2, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: A precisão para esta função macro é limitada a valores de número inteiro menores que 2^{24} . Nenhum valor negativo é permitido.

Nota: O operador BIT_AND pode ser abreviado com um e comercial (&). Use o e comercial para separar os dois argumentos (por exemplo, para especificar BIT_AND(V1, 3), é possível simplesmente digitar V1&3.

Exemplos

TEMP = 3 BIT_AND 7 ou TEMP = 3 & 7 Cria uma nova coluna denominada TEMP contendo o valor três (AND bit a bit de 011 e 111 igual a 011).
TEMP = V1 & 8 Cria uma nova coluna denominada TEMP, em que cada valor é o conteúdo da coluna V1, com o AND bit a bit executado com o valor binário 1000.
TEMP = V1 & V1 Cria uma nova coluna denominada TEMP possuindo o mesmo conteúdo que a coluna V1 (cada valor de AND executado com si próprio produz ele mesmo).
TEMP = V1 & V2 Cria uma nova coluna denominada TEMP, em que cada valor é o valor da linha da coluna V1 com AND bit a bit executado com o valor da linha correspondente da coluna V2.
TEMP = V1:V3 & V4:V6 Cria três novas colunas denominadas TEMP, VX e VY. A coluna TEMP contém os valores em V1 com o AND bit a bit executado com os valores da linha correspondente da coluna V4. A coluna VX contém os valores de AND bit a bit executado das colunas V2 e V5. A coluna VY contém os valores de AND bit a bit executado das colunas V3 e V6.

`TEMP = V1[10:20] & V2` ou `TEMP = V1[10:20] & V2[1:11]` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm o resultado de AND bit a bit executado dos valores nas linhas 10-20 da coluna `V1` pelos valores nas linhas 1-11 da coluna `V2`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`BIT_NOT`	Calcula o NOT bit a bit do conteúdo do intervalo de dados especificado
`BIT_OR`	Calcula o OR bit a bit entre dois intervalos de dados especificados
`BIT_XOR` ou `XOR`	Calcula o XOR bit a bit entre dois intervalos de dados especificados

Macro BIT_NOT

O `BIT_NOT` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
BIT_NOT data ~ data
```

Parâmetros

`data`

Os números inteiros não negativos para NOT bit a bit. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`BIT_NOT` calcula o NOT bit a bit dos valores no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo o NOT bit a bit dos valores nas colunas correspondentes de `data`.

Nota: A precisão para esta função macro é limitada a valores de número inteiro menores que 2^{24} . Nenhum valor negativo é permitido.

Nota: Usar uma coluna contendo o mesmo número *x* em cada linha como `data` é o mesmo que utilizar a constante *x* como `data`.

Nota: O operador `BIT_NOT` pode ser abreviado com um til (~). Use o til antes do valor dos dados (por exemplo, para especificar `BIT_NOT(V1)`, é possível simplesmente digitar `~V1`.

Exemplos

`TEMP = BIT_NOT 3` ou `TEMP = ~3` Cria uma nova coluna denominada `TEMP` contendo o valor quatro (NOT bit a bit de `011` igual a `100`).
`TEMP = ~V1` Cria uma nova coluna denominada `TEMP`, em que cada valor é o NOT bit a bit do conteúdo da coluna `V1`.
`TEMP = ~V1:V3` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os NOT bit a bit do conteúdo da coluna `V1`, os valores da coluna `VX` são os NOT bit a bit do conteúdo da coluna `V2`, e os valores da coluna `VY` são os NOT bit a bit do conteúdo da coluna `V3`.
`TEMP = ~V1[100:200]` Cria uma nova coluna denominada `TEMP`, em que as 101 primeiras células contêm os NOT bit a bit dos valores nas linhas 1-50 da coluna `V1`.

Funções Relacionadas

Função	Descrição
`BIT_AND`	Calcula o AND bit a bit entre dois intervalos de dados especificados
`BIT_OR`	Calcula o OR bit a bit entre dois intervalos de dados especificados
`BIT_XOR` ou `XOR`	Calcula o XOR bit a bit entre dois intervalos de dados especificados

Macro BIT_OR

O `BIT_OR` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
data1 BIT_OR data2 data1 OR data2 data1 | data2
```

Parâmetros

`data1`

Os números inteiros não negativos para executar OR bit a bit com os valores em `data2`. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`data2`

Um ou mais números inteiros não negativos para executar OR bit a bit com os valores em `data1`. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em `data2` deve ser igual ao número de colunas em `data1`, a menos que

data2 seja uma constante. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

BIT_OR executa um OR bit a bit entre dois intervalos de dados especificados. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em data1 com OR bit a bit executado com a coluna correspondente de data2 (ou seja, a primeira coluna de data1 tem o OR bit a bit executado com a primeira coluna de data, a segunda coluna com a segunda coluna, e assim por diante).

Se data2 for uma constante, cada valor em data1 terá o OR bit a bit executado por esse valor. Se data2 contiver uma ou mais colunas, os cálculos serão executados em uma base de linha por linha entre uma coluna data2 e uma coluna de data2. A primeira linha de data1 tem o OR bit a bit executado para o valor da primeira linha de data2, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: A precisão para esta função macro é limitada a valores de número inteiro menores que 2^{24} . Nenhum valor negativo é permitido.

Nota: O operador BIT_OR pode ser abreviado com uma barra vertical (|). Use a barra vertical para separar as duas colunas (por exemplo, para especificar BIT_OR(V1, 3), é possível simplesmente digitar V1|3. Também é possível usar OR.

Exemplos

TEMP = 3 BIT_OR 7 ou TEMP = 3 OR 7 ou TEMP = 3 | 7 Cria uma nova coluna denominada TEMP contendo o valor sete (OR bit a bit de 011 e 111 igual a 111).
TEMP = V1 | 8 Cria uma nova coluna denominada TEMP, em que cada valor é o conteúdo da coluna V1, com o OR bit a bit executado com o valor binário 1000.
TEMP = V1 | V1 Cria uma nova coluna denominada TEMP possuindo o mesmo conteúdo que a coluna V1 (cada valor de OR executado com si próprio produz ele mesmo).
TEMP = V1 | V2 Cria uma nova coluna denominada TEMP, em que cada valor é o valor da linha da coluna V1 com OR bit a bit executado com o valor da linha correspondente da coluna V2.
TEMP = V1:V3 | V4:V6 Cria três novas colunas denominadas TEMP, VX e VY. A coluna TEMP contém os valores em V1 com OR definido logicamente com os valores da linha correspondente da coluna V4. A coluna VX contém os valores de OR definidos logicamente das colunas V2 e V5. A coluna VY contém os valores de OR definidos logicamente das colunas V3 e V6.

`TEMP = V1[10:20] | V2` ou `TEMP = V1[10:20] | V2[1:11]` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm o resultado de OR bit a bit executado dos valores nas linhas 10-20 da coluna `V1` pelos valores nas linhas 1-11 da coluna `V2`. Outras células em `TEMP` estão vazias.

Funções Relacionadas

Função	Descrição
`BIT_AND`	Calcula o AND bit a bit entre dois intervalos de dados especificados
`BIT_NOT`	Calcula o NOT bit a bit do conteúdo do intervalo de dados especificado
`BIT_XOR` ou `XOR`	Calcula o XOR bit a bit entre dois intervalos de dados especificados

Macro BIT_XOR

O `BIT_XOR` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
data1 BIT_XOR data2
```

Parâmetros

`data1`

Os números inteiros não negativos para executar XOR bit a bit com os valores em `data2`. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`data2`

Um ou mais números inteiros não negativos para executar XOR bit a bit com os valores em `data1`. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em `data2` deve ser igual ao número de colunas em `data1`, a menos que `data2` seja uma constante. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`BIT_XOR` executa um XOR bit a bit entre dois intervalos de dados especificados. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em `data1` com XOR bit a bit executado com a coluna correspondente de `data2` (ou seja, a primeira coluna de `data1` tem o XOR bit a bit executado com a primeira coluna de `data`, a segunda coluna com a segunda coluna, e assim por diante).

Se data2 for uma constante, cada valor em data1 terá o XOR bit a bit executado por esse valor. Se data2 contiver uma ou mais colunas, os cálculos serão executados em uma base de linha por linha entre uma coluna data2 e uma coluna de data2. A primeira linha de data1 tem o XOR bit a bit executado com o valor da primeira linha de data2, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: A precisão para esta função macro é limitada a valores de número inteiro menores que 2^{24} . Nenhum valor negativo é permitido.

Exemplos

`TEMP = 3 BIT_XOR 7` Cria uma nova coluna denominada TEMP contendo o valor quatro (XOR bit a bit de 011 e 111 igual a 100).
`TEMP = V1 BIT_XOR 8` Cria uma nova coluna denominada TEMP, em que cada valor é o conteúdo da coluna V1, com o XOR bit a bit executado com o valor binário 1000.
`TEMP = V1 BIT_XOR V1` Cria uma nova coluna denominada TEMP contendo todos zeros (cada valor de XOR executado com ele mesmo produz zero).
`TEMP = V1 BIT_XOR V2` Cria uma nova coluna denominada TEMP, em que cada valor é o valor da linha da coluna V1 com XOR bit a bit executado com o valor da linha correspondente da coluna V2.
`TEMP = V1:V3 BIT_XOR V4:V6` Cria três novas colunas denominadas TEMP, VX e VY. A coluna TEMP contém os valores em V1 com o XOR bit a bit executado com os valores da linha correspondente da coluna V4. A coluna VX contém os valores de XOR bit a bit executado das colunas V2 e V5. A coluna VY contém os valores de XOR bit a bit executado das colunas V3 e V6.
`TEMP = V1[10:20] BIT_XOR V2` ou `TEMP = V1[10:20] BIT_XOR V2[1:11]` Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm o resultado de XOR bit a bit executado dos valores nas linhas 10-20 da coluna V1 pelos valores nas linhas 1-11 da coluna V2. Outras células em TEMP estão vazias.

Funções relacionadas

Função	Descrição
BIT_AND	Calcula o AND bit a bit entre dois intervalos de dados especificados
BIT_NOT	Calcula o NOT bit a bit do conteúdo do intervalo de dados especificado
BIT_OR	Calcula o OR bit a bit entre dois intervalos de dados especificados

Macro CEILING

O CEILING está disponível apenas no IBM Campaign.

Sintaxe

```
CEILING(data)
```

Parâmetros

`data`

Os valores numéricos dos quais calcular o limite. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`CEILING` Calcula o limite dos valores no intervalo de dados especificado. O limite de um número é o menor número inteiro *não* menor que o número. `CEILING` retorna uma nova coluna para cada coluna de entrada, cada uma contendo o limite de números na coluna de entrada correspondente.

Exemplos

`TEMP = CEILING(4.3)` Cria uma nova coluna denominada `TEMP` contendo o valor `5`.
`TEMP = CEILING(2.9)` Cria uma nova coluna denominada `TEMP` contendo o valor `-2`.
`TEMP = CEILING(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é o limite do conteúdo da coluna `V1`.
`TEMP = CEILING(V1:V3)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os limites do conteúdo da coluna `V1`, os valores da coluna `VX` são os limites do conteúdo da coluna `V2`, e os valores da coluna `VY` são os limites do conteúdo da coluna `V3`.
`TEMP = CEILING(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os limites dos valores nas linhas 10-20 da coluna `V1`. Outras células em `TEMP` estão vazias.
`TEMP = CEILING(V1[50:99]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-50 (as outras células estão vazias). Os valores na coluna `TEMP` são os limites das linhas da coluna `V1`, e os valores na coluna `VX` são os limites dos valores na coluna `V2`.

Funções relacionadas

Função	Descrição
`FLOOR` ou `INT`	Calcula o piso de cada valor no intervalo de dados especificado.
`FRACTION`	Retorna a parte fracionária de cada valor no intervalo de dados especificado.

Função	Descrição
TRUNCATE	Retorna a parte não fracionária de cada valor no intervalo de dados especificado.

Macro COLUMN

O COLUMN está disponível apenas no IBM Campaign.

Sintaxe

```
COLUMN(data [, data]...) ou (data [, data]...)
```

Parâmetros

data

Um valor para uso na criação de uma coluna. Isso pode ser um valor constante (numérico ou de texto ASCII entre aspas), uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Esse parâmetro pode ser repetido várias vezes, mas os parâmetros subsequentes devem ter a mesma dimensão (ou seja, largura da coluna) que o primeiro parâmetro. Todos os valores em todos os parâmetros data devem ser numéricos ou texto ASCII (ou seja, não é possível combinar valores numéricos e de texto). Se diversos parâmetros data forem fornecidos, todos eles deverão ter o mesmo número de colunas. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

COLUMN concatena verticalmente suas entradas em colunas de um grupo de função. Ele retorna o mesmo número de colunas novas que é fornecido em cada parâmetro de entrada. Um número ilimitado de argumentos pode ser fornecido. Todos os argumentos devem ser numéricos ou sequências de texto ASCII (ou seja, não é possível combinar valores numéricos e de texto).

Nota: A função macro COLUMN pode ser abreviada ao listar os argumentos data separados por vírgulas dentro de parênteses (por exemplo, TEMP = MEAN((1,2,3,4), ALL)). Se não for usado dentro de outra função macro, o par de parênteses não será necessário (por exemplo, V1=1,2,3 é equivalente a V1=COLUMN(1,2,3)).

Exemplos

```
TEMP = COLUMN(3, 4, 5) ou TEMP = (3,4,5) ou TEMP = 3,4,5
```

Cria uma nova coluna denominada TEMP com as três primeiras células contendo os valores 3, 4 e 5.

```
TEMP = COLUMN("one","two", "three")
```

Cria uma nova coluna denominada TEMP com as três primeiras células contendo os valores "one ", "two " e "three ".

```
TEMP = AVG(V1), STDV(V1)
```

Cria uma nova coluna denominada TEMP com a média da coluna V1 na primeira célula e o desvio padrão da coluna V1 na segunda célula.

`TEMP = V1:V2, V3:V4` Cria duas novas colunas denominadas `TEMP` e `VX`, em que a coluna `TEMP` contém os valores da coluna `V1`, seguido pelos valores da coluna `V3`. A coluna `VX` contém os valores da coluna `V2`, seguido pelos valores da coluna `V4`.
`TEMP = V1:V2, V3:V4` Cria duas novas colunas denominadas `TEMP` e `VX`, em que a coluna `TEMP` contém os valores das células 1-10 da coluna `V1`, seguido pelos valores da coluna `V3`. A coluna `VX` contém os valores das células 1-10 da coluna `V2`, seguido por todos os valores da coluna `V4`.
`TEMP = V1:V2, V3:V4` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma contendo um valor único. A coluna `TEMP` contém a média das colunas `V1` e `V2`. A coluna `VX` contém a média das colunas `V3` e `V4`.

Macro COS

O `COS` está disponível apenas no IBM Campaign.

Sintaxe

```
COS(data [, units_keyword])
```

Parâmetros

`data`

Os valores numéricos dos quais calcular o cosseno. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`units_keyword`

Essa palavra-chave opcional determina se os valores de entrada e os resultados são interpretados como graus ou radianos. Selecione uma das opções a seguir:

`RADIAN` - Executa os cálculos em radianos (padrão)

`DEGREE` - Executa o cálculo em graus

Se esse parâmetro não for especificado, o padrão será radianos. (Para converter de radianos em graus, divida pelo `PI` e multiplique por 180).

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte “Especificações de Formato” na página 9.

Descrição

`COS` calcula o cosseno dos valores no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo o cosseno dos números na coluna de entrada correspondente.

Exemplos

`TEMP = COS(PI)` ou `TEMP = COS(PI, 0)` ou `TEMP = COS(PI, RADIAN)` Retorna uma nova coluna denominada `TEMP` contendo um valor único de `-1`.
`TEMP = COS(90, 1)` ou `TEMP = COS(90, DEGREE)` Cria uma nova coluna denominada `TEMP` contendo um valor único de zero.
`TEMP = COS(V1)` ou `TEMP = COS(V1, 0)` ou `TEMP = COS(V1, RADIAN)` Cria uma nova coluna denominada `TEMP`, em que cada valor é o cosseno (em radianos) do conteúdo da coluna `V1`.
`TEMP = COS(V1:V3, 1)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os cossenos do conteúdo da coluna `V1`, os valores da coluna `VX` são os cossenos do conteúdo da coluna `V2`, e os valores da coluna `VY` são os cossenos do conteúdo da coluna `V3`. Todos os valores estão em graus.
`TEMP = COS(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os cossenos dos valores nas linhas 10-20 da coluna `V1` (em radianos). Outras células em `TEMP` estão vazias.
`TEMP = COS(V1[1:5]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são os cossenos das linhas correspondentes da coluna `V1`, e os valores na coluna `VX` são os cossenos das linhas correspondentes da coluna `V2`. Todos os valores estão em radianos.

Funções relacionadas

Função	Descrição
`ACOS`	Calcula o arco cosseno do conteúdo do intervalo de dados especificado
`COSH`	Calcula o cosseno hiperbólico do conteúdo do intervalo de dados especificado
`SIN`	Calcula o seno do conteúdo do intervalo de dados especificado
`TAN`	Calcula a tangente do conteúdo do intervalo de dados especificado

Macro COSH

O `COSH` está disponível apenas no IBM Campaign.

Sintaxe

```
COSH(data [, units_keyword])
```

Parâmetros

`data`

Os valores numéricos dos quais calcular o cosseno hiperbólico. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`units_keyword`

Essa palavra-chave opcional determina se os valores de entrada e os resultados são interpretados como graus ou radianos. Selecione uma das opções a seguir:

`RADIAN` - Executa os cálculos em radianos (padrão)

`DEGREE` - Executa o cálculo em graus

Se esse parâmetro não for especificado, o padrão será radianos. (Para converter de radianos em graus, divida pelo `PI` e multiplique por 180).

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte “Especificações de Formato” na página 9.

Descrição

`COSH` calcula o cosseno hiperbólico dos valores no intervalo de dados especificado. Para o *x* em radianos, o cosseno hiperbólico de um número é:

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

em que *e* é o número natural, 2,7182818. `COSH` retorna uma nova coluna para cada coluna de entrada, cada uma contendo o cosseno hiperbólico dos números na coluna de entrada correspondente.

Nota: Se o valor de *x* for muito grande, um erro de estouro será retornado. Isso ocorrerá se o *cosh* (*x*) exceder o valor de vírgula flutuante máximo de 32 bits.

Exemplos

`TEMP = COSH(0)` ou `TEMP = COSH(0, 0)` ou `TEMP = COSH(0, RADIAN)` Cria uma nova coluna denominada `TEMP` contendo o valor um.
`TEMP = COSH(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é o cosseno hiperbólico (em radianos) do conteúdo da coluna `V1`.
`TEMP = COSH(V1:V3, 1)` ou `TEMP = COSH(V1:V3, DEGREE)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os cossenos hiperbólicos do conteúdo da coluna `V1`, os valores da coluna `VX` são os cossenos hiperbólicos do conteúdo da coluna `V2`, e os valores da coluna `VY` são os cossenos hiperbólicos do conteúdo da coluna `V3`. Todos os valores estão em graus.

`TEMP = COSH(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os cossenos hiperbólicos dos valores nas linhas 10-20 da coluna `V1` (em radianos). Outras células em `TEMP` estão vazias.
`TEMP = COSH(V1[1:5]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são os cossenos hiperbólicos das linhas correspondentes da coluna `V1`, e os valores na coluna `VX` são os cossenos hiperbólicos das linhas correspondentes da coluna `V2`. Todos os valores estão em radianos.

Funções relacionadas

Função	Descrição
`ACOS`	Calcula o arco cosseno do conteúdo do intervalo de dados especificado
`COS`	Calcula o cosseno do conteúdo do intervalo de dados especificado
`SINH`	Calcula o seno hiperbólico do conteúdo do intervalo de dados especificado
`TANH`	Calcula a tangente hiperbólica do conteúdo do intervalo de dados especificado

Macro COT

O `COT` está disponível apenas no IBM Campaign.

Sintaxe

```
COT(data [, units_keyword])
```

Parâmetros

`data`

Os valores numéricos dos quais calcular a cotangente. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`units_keyword`

Essa palavra-chave opcional determina se os valores de entrada e os resultados são interpretados como graus ou radianos. Selecione uma das opções a seguir:

`RADIAN` - Executa os cálculos em radianos (padrão)

`DEGREE` - Executa o cálculo em graus

Se esse parâmetro não for especificado, o padrão será radianos. (Para converter de radianos em graus, divida pelo `PI` e multiplique por 180).

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte “Especificações de Formato” na página 9.

Descrição

COT calcula a cotangente dos valores no intervalo de dados especificado. A cotangente é o recíproco da tangente. COT retorna uma nova coluna para cada coluna de entrada, cada uma contendo a cotangente dos números na coluna de entrada correspondente.

Nota: Se uma célula contiver um valor cuja tangente é zero, então o arco cotangente será infinito. Nesse caso, COT retornará o maior número de vírgula flutuante de 32 bits.

Exemplos

`TEMP = COT(90)` ou `TEMP = COT(90, 0)` ou `TEMP = COT(90, RADIAN)` Retorna uma nova coluna denominada TEMP contendo o valor -0.5.
`TEMP = COT(0)` Retorna uma nova coluna denominada TEMP contendo o valor MAX_FLOAT_32.
`TEMP = COT(V1, 1)` ou `TEMP = COT(V1, DEGREE)` Cria uma nova coluna denominada TEMP, em que cada valor é a cotangente do conteúdo (em graus) da coluna V1.
`TEMP = COT(V1:V3, 1)` Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são as cotangentes do conteúdo da coluna V1, os valores da coluna VX são as cotangentes do conteúdo da coluna V2, e os valores da coluna VY são as cotangentes do conteúdo da coluna V3. Todos os valores estão em graus.
`TEMP = COT(V1[10:20])` Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm as cotangentes dos valores nas linhas 10-20 da coluna V1 (em radianos). Outras células em TEMP estão vazias.
`TEMP = COT(V1[1:5]:V2)` Cria duas novas colunas denominadas TEMP e VX, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna TEMP são as cotangentes das linhas correspondentes da coluna V1, e os valores na coluna VX são as cotangentes das linhas correspondentes da coluna V2. Todos os valores estão em radianos.

Funções relacionadas

Função	Descrição
ACOT	Calcula a arco cotangente do conteúdo do intervalo de dados especificado
COS	Calcula o cosseno do conteúdo do intervalo de dados especificado
SIN	Calcula o seno do conteúdo do intervalo de dados especificado
TAN	Calcula a tangente do conteúdo do intervalo de dados especificado

Macro COUNT

O COUNT está disponível apenas no IBM Campaign.

Sintaxe

```
COUNT(data)
```

Parâmetros

data

O intervalo de células no qual contar o número de células. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

COUNT conta o número de valores no intervalo de dados especificado. Ele retorna uma nova coluna contendo um valor único representando o número de células contendo valores no intervalo de dados especificado.

Nota: Contar uma coluna em branco retorna zero.

Exemplos

`TEMP = COUNT(AVG(V1:V5))` Cria uma nova coluna denominada TEMP contendo um valor único de um (a função AVG retorna uma única célula no modo padrão).
`TEMP = COUNT(V1)` Cria uma nova coluna denominada TEMP contendo um valor único indicando o número de células contendo valores na coluna V1.
`TEMP = COUNT(V1:V3)` Cria uma nova coluna denominada TEMP contendo um valor único indicando o número de células contendo valores nas colunas V1, V2 e V3.
`TEMP = COUNT(V1[10:20])` Cria uma nova coluna denominada TEMP contendo o valor 11 (os intervalos são inclusivos), dado que a todas as células contêm valores.
`TEMP = COUNT(V1[1:5]:V4)` Cria uma nova coluna denominada TEMP contendo o valor 20 (5 células em cada coluna vezes 4 colunas = 20 células), dado que todas as células contêm valores.
`TEMP = COUNT(V1[1:10])` Cria uma nova coluna denominada TEMP contendo o valor 3, dado que as linhas 1-3 da coluna V1 contêm valores e as linhas 4-10 estão vazias.

Funções relacionadas

Função	Descrição
SUM ou TOTAL	Calcula a soma de um intervalo de células

Macro CURRENT_DATE

O CURRENT_DATE está disponível no IBM Campaign e no IBM Interact.

Sintaxe

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

Parâmetros

formato

Uma das palavras-chave na tabela a seguir especificando o formato de data de date_string.

Nota: Consulte "Palavras-chave de Formato Válido" para obter mais informações sobre formatos de data disponíveis.

Descrição

CURRENT_DATE retorna a data atual no format. A data é determinada pelo relógio no servidor do IBM . Se nenhuma palavra-chave format for fornecida, o padrão de DELIM_M_D_Y será usado.

Para todos os bancos de dados recomendados, o IBM Campaign tenta executar a macro CURRENT_DATE no banco de dados utilizando uma chamada de SQL de horário atual suportada pelo banco de dados (por exemplo, SYSDATE, GETDATE, DATE ou TODAY). Nestes casos, todos os parâmetros (incluindo o formato da data) desta função de macro são ignorados e a saída inclui tudo o que é retornado pelo banco de dados (por exemplo, um componente de tempo pode ser incluído na saída). Se isso ocorrer e você desejar retornar apenas a data ou a data em um formato diferente, será possível gravar sua própria macro customizada utilizando SQL bruta ou usar outras macros do IBM . Por Exemplo:

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

Em alguns casos, a macro CURRENT_DATE() é executada no servidor IBM Campaign (por exemplo, se estiver executando com relação a um arquivo simples, a um banco de dados não recomendado sem suporte SQL equivalente ou se a expressão de macro do Campaign não puder ser resolvida no banco de dados). Nestes casos, todos os parâmetros são reconhecidos e a saída é retornada no formato selecionado.

Nota: Nem todos os formatos disponíveis no IBM Campaign são suportados pelo IBM Interact.

Exemplos

Se a data de hoje for o dia 7 de setembro de 2000, CURRENT_DATE() retornará "07/09/2000".

Funções relacionadas

Função	Descrição
DATE_FORMAT	Converte datas de um formato para outro.
DATE_JULIAN	Retorna a data Juliana da entrada.
DATE_STRING	Retorna a sequência de datas da data Juliana.
DATE	Converte uma sequência de datas em data Juliana.

Macro CURRENT_DAY

O CURRENT_DAY está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
CURRENT_DAY()
```

Descrição

CURRENT_DAY retorna o dia atual do mês como um número entre 1-31. A data é determinada pelo relógio do sistema no servidor IBM .

Exemplos

Se a data de hoje for o dia 19 de junho, CURRENT_DAY() retornará o número 19.

Funções relacionadas

Função	Descrição
CURRENT_JULIAN	Retorna o número Juliano para a data atual.
CURRENT_MONTH	Retorna o mês atual como um número.
CURRENT_TIME	Retorna a hora atual como uma sequência.
CURRENT_WEEKDAY	Retorna o dia da semana atual como um número.
CURRENT_YEAR	Retorna o ano atual como um número.

Macro CURRENT_JULIAN

O CURRENT_JULIAN está disponível apenas no IBM Campaign.

Sintaxe

```
CURRENT_JULIAN()
```

Descrição

CURRENT_JULIAN converte um número juliano na data atual (o número de dias decorridos desde 1º de janeiro de 0000). Isto é equivalente à macro DATE(CURRENT_DATE()).

Exemplos

Se a data de hoje for 31 de agosto de 2000, CURRENT_JULIAN() retornará o número 730729.

Funções relacionadas

Função	Descrição
CURRENT_DAY	Retorna o dia atual como um número.
CURRENT_MONTH	Retorna o mês atual como um número.
CURRENT_TIME	Retorna a hora atual como uma sequência.
CURRENT_WEEKDAY	Retorna o dia da semana atual como um número.
CURRENT_YEAR	Retorna o ano atual como um número.

Macro CURRENT_MONTH

O CURRENT_MONTH está disponível no IBM Campaign e no IBM Interact.

Sintaxe

CURRENT_MONTH()

Descrição

CURRENT_MONTH retorna o mês atual do ano como um número entre 1-12.

Exemplos

Se a data de hoje for o dia 19 de junho, CURRENT_MONTH() retornará o número 6.

Funções relacionadas

Função	Descrição
CURRENT_DAY	Retorna o dia atual como um número.
CURRENT_JULIAN	Retorna o número Juliano atual.
CURRENT_TIME	Retorna a hora atual como uma sequência.
CURRENT_WEEKDAY	Retorna o dia da semana atual como um número.
CURRENT_YEAR	Retorna o ano atual como um número.

Macro CURRENT_TIME

The CURRENT_TIME está disponível apenas no IBM Campaign.

Sintaxe

CURRENT_TIME()

Descrição

CURRENT_TIME retorna a hora atual como uma sequência. O horário é determinado pelo relógio do sistema no servidor do IBM .

Configuração de data em seu aplicativo da web

Para exibir corretamente as datas em seu aplicativo da web em versões atuais do IBM Campaign, seu arquivo de configuração do servidor de backend deve primeiro ser configurado corretamente. Isso é importante principalmente para os parâmetros dDateFormat e DateOutputFormatString do banco de dados que contém as tabelas de sistema. Se eles não forem configurados corretamente, as datas também serão exibidas incorretamente no Campaign. Configure essas propriedades utilizando o IBM Marketing Platform.

Para configurar datas para um idioma específico em seu aplicativo da web

Nota: Todos os arquivos referenciados são instalados pelo instalador do aplicativo da web, a menos que especificamente observado.

Importante: O webapphome refere-se ao diretório no qual o aplicativo da web Campaign foi instalado. language_code se refere a uma ou mais configurações de idioma que você escolher para seu sistema.

1. Edite o arquivo webapphome/conf/campaign_config.xml para assegurar que o language_code esteja presente na lista separada por vírgula na tag <supportedLocales>, conforme mostrado abaixo:

   ```
   <supportedLocales>en_US, language_code</supportedLocales>
   ```

2. No diretório webapphome/webapp, copie a árvore de diretórios inteira en_US to language_code (com distinção entre maiúsculas e minúsculas).
3. Em webapphome/webapp/WEB-INF/classes/resources, copie StaticMessages_en_US.properties para StaticMessages_ language_code.properties. Além disso, copie ErrorMessages_en_US.properties para ErrorMessages_ language_code.properties.
4. Edite o StaticMessages_ language_code.properties: procure por DatePattern e altere-o para que leia DatePattern=dd/MM/yyyy (com distinção entre maiúsculas e minúsculas).

   Nota: Esse formato é definido por Java™. Detalhes completos sobre o formato podem ser localizados na documentação Java para java.text.SimpleDateFormat em http://java.sun.com. O arquivo StaticMessages.properties não precisa ser modificado.
5. Para WebSphere: Crie novamente o arquivo jar do aplicativo da web.
6. Para WebLogic: Remova o módulo do aplicativo da web atual.
   a. Inclua o novo módulo.
   b. Reimplemente o aplicativo da web.
   c. Não é necessário reiniciar o listener do Campaign.
7. Assegure-se de que a configuração de idioma do navegador da web tenha language_code configurado para a primeira prioridade. Para obter mais detalhes, consulte as seções a seguir para configurar seu navegador da web para o idioma correto e para configurar seu computador para exibir um idioma específico.

Nota: Assegure-se de utilizar um hífen e não um sublinhado em `language_code`. A configuração do aplicativo da web é o único local onde um hífen é usado ao invés de um sublinhado.

8. Efetue login no Campaign. As datas devem ser exibidas no Campaign no formato especificado em `StaticMessages_language_code.properties`.

Para obter informações sobre como configurar a hora para o IBM Campaign, consulte a documentação do *IBM Campaign*.

Exemplos

Se o horário for 10h54, o `CURRENT_TIME()` retornará a sequência "10:54:00 AM".

Funções Relacionadas

Função	Descrição
`CURRENT_DAY`	Retorna o dia atual como um número.
`CURRENT_JULIAN`	Retorna o número Juliano atual.
`CURRENT_WEEKDAY`	Retorna o dia da semana atual como um número.
`CURRENT_YEAR`	Retorna o ano atual como um número.

Macro CURRENT_WEEKDAY

O `CURRENT_WEEKDAY` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
CURRENT_WEEKDAY()
```

Descrição

`CURRENT_WEEKDAY` retorna o dia atual da semana como um número entre 0-6. Domingo é representado como 0, segunda-feira, como 1, e assim por diante.

Exemplos

Se hoje for sexta-feira, `CURRENT_WEEKDAY()` retornará o número 5.

Funções relacionadas

Função	Descrição
`CURRENT_DAY`	Retorna o dia atual como um número.
`CURRENT_JULIAN`	Retorna o número Juliano atual.
`CURRENT_MONTH`	Retorna o mês atual como um número.
`CURRENT_TIME`	Retorna a hora atual como uma sequência.
`CURRENT_YEAR`	Retorna o ano atual como um número.

Macro CURRENT_YEAR

O `CURRENT_YEAR` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
CURRENT_YEAR()
```

Descrição

CURRENT_YEAR retorna o ano atual como um número.

Exemplos

Se o ano atual for 2000, CURRENT_YEAR() retornará o número: 2000.

Funções relacionadas

Função	Descrição
CURRENT_DAY	Retorna o dia atual como um número.
CURRENT_JULIAN	Retorna o número Juliano atual.
CURRENT_MONTH	Retorna o mês atual como um número.
CURRENT_TIME	Retorna a hora atual como uma sequência.
CURRENT_WEEKDAY	Retorna o dia da semana atual como um número.
MONTHOF	Retorna o mês do ano como um número.
WEEKDAYOF	Retorna o dia da semana como um número.
YEAROF	Retorna o ano como um número.

Macro DATE

O DATE está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
DATE(date_string [, format])
```

Parâmetros

date_string

Texto que representa uma data válida.

formato

Uma das palavras-chave na tabela em "Palavras-Chave de Formatos Válidos" na página 17, especificando o formato de data do date_string.

Descrição

DATE converte uma sequência de data em uma data Juliana (o número de dias decorridos desde 1º de janeiro de 0000). Praticamente, nenhum formato de data é suportado por meio da palavra-chave format opcional, que especifica como a data é representada. Se nenhuma palavra-chave format for fornecida, o padrão de DELIM_M_D_Y será usado.

Os formatos de data são de largura fixa (por exemplo, a data 28 de fevereiro de 1970 é representada como 02281970 no formato MMDDYYYY), ou delimitados (por exemplo, 28 de fevereiro de 1970, 28-2-1970 ou 28/02/1970). Todos os exemplos anteriores são variantes do formato `DELIM_M_D_YY`.

Em formatos delimitados, os delimitadores são barra (/), traço(-), espaço (), vírgula (,) ou dois pontos (:); os anos podem ser representados por 2 ou 4 dígitos; e os meses podem ser escritos por extenso, (por exemplo, Fevereiro), abreviados (por exemplo, Fev) ou numéricos (por exemplo, 2 ou 02).

Para todos os anos especificado como dois dígitos:

- Os anos de dois dígitos menores que o limite do Ano 2000 (o padrão é 20, mas pode ser configurado pelo usuário) são consideradas como estando nos anos 2000.
- Anos de dois dígitos maiores ou iguais ao limite são consideradas como estando nos anos de 1900.

Nota: Nem todos os formatos disponíveis no IBM Campaign são suportados pelo IBM Interact.

Exemplos

`DATE("8/31/2000")` retorna o número `730729`.

Funções relacionadas

Função	Descrição
`DATE_FORMAT`	Converte datas de um formato para outro.
`DATE_JULIAN`	Retorna a data Juliana da entrada.
`DATE_STRING`	Retorna a sequência de datas da data Juliana.
`CURRENT_DATE`	Retorna a data atual em um formato especificado.

Macro DATE_FORMAT

O `DATE_FORMAT` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
DATE_FORMAT(date_string, input_format, output_format)
```

Parâmetros

`date_string`

Um texto representando uma data válida.

`input_format`

Uma das palavras-chave na tabela abaixo especificando o formato de data de `date_string`.

`output_format`

Uma das palavras-chave na tabela abaixo especificando o formato de data de saída desejado.

Descrição

DATE_FORMAT() transforma uma data de input_format em outro formato output_format.

Se a data for de largura fixa, ela deverá ser configurada para um dos seguintes valores:

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

MM é um mês de 2 dígitos e MMM é a abreviação do mês de 3 caracteres.

Se a data for delimitada (qualquer delimitador pode ser utilizado, incluindo SPACE, DASH, SLASH), ela deverá ser configurada para um destes valores:

- DELIM_D_M_Y
- DELIM_M_D_Y
- DELIM_Y_M_D

Nota: Nem todos os formatos disponíveis no IBM Campaign são suportados pelo IBM Interact.

Exemplos

DATE_FORMAT("012171", MMDDYY, MMDDYYYY) retorna a sequência "01211971".

Nota: Consulte “Macro DATE” na página 51 para obter informações adicionais sobre formatos de data válidos.

Funções Relacionadas

Função	Descrição
DATE	Converte uma sequência de datas em uma data Juliana.
DATE_JULIAN	Retorna a data Juliana da entrada.
DATE_STRING	Retorna a sequência de datas da data Juliana.

Macro DATE_JULIAN

O DATE_JULIAN está disponível apenas no IBM Campaign.

Sintaxe

DATE_JULIAN(year, month, day)

Parâmetros

`year`

Número de ano de 2 ou 4 dígitos válido.

`month`

Número do mês válido entre 1-12.

`day`

Número de dia válido entre 1-31.

Descrição

`DATE_JULIAN` retorna a data Juliana da entrada especificada. A data juliana é o número de dias decorridos desde 1º de janeiro de 0000.

Exemplos

DATE_JULIAN (`2000,08,31`) retorna o número 730729.

Funções relacionadas

Função	Descrição
`DATE`	Converte uma sequência de datas em uma data Juliana.
`DATE_FORMAT`	Converte datas de um formato para outro.
`DATE_STRING`	Retorna a sequência de datas da data Juliana.

Macro DATE_STRING

O `DATE_STRING` está disponível apenas no IBM Campaign.

Sintaxe

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

Parâmetros

`julian_date`

Um número que representa uma data Juliana, o número de dias decorridos desde 1º de janeiro de 0000.

`output_format`

Sequência, em formato de data válido.

`max_length`

`format_string`

Uma sequência de formatação, opcionalmente incluindo qualquer combinação dos códigos de formato a seguir:

Código	Descrição
`%a`	Nome do dia da semana abreviado
`%A`	Nome do dia da semana completo
`%b`	Nome do mês abreviado
`%B`	Nome do mês completo
`%c`	Representação de data e hora apropriada para o código do idioma
`%d`	Dia do mês (01 – 31)
`%H`	Hora no formato de 24 horas (00 – 23)
`%I`	Hora no formato de 12 horas (01 – 12)
`%j`	Dia do ano (001 – 366)
`%m`	Mês (01 – 12)
`%M`	Minuto (00 - 59)
`%p`	Indicador AM/PM do código de idioma atual para o relógio de 12 horas
`%S`	Segundo (00 - 59)
`%U`	Semana do ano, com Domingo sendo o primeiro dia da semana (00 – 51)
`%w`	Dia da semana (0 – 6; Domingo é 0)
`%W`	Semana do ano, com Segunda-feira sendo o primeiro dia da semana (00 – 51)
`%x`	Representação de data para o código do idioma atual
`%X`	Representação de hora para o código do idioma atual
`%y`	Ano de 2 dígitos (00 – 99)
`%Y`	Ano de 4 dígitos
`%z, %Z`	Nome ou abreviação de fuso horário; nenhuma saída se o fuso horário for desconhecido
`%%`	Sinal de porcentagem

Descrição

`DATE_STRING` retorna a sequência de datas da data Juliana. Se `output_format` não for fornecido, a palavra-chave padrão `DELIM_M_D_Y` será utilizada.

Exemplos

`DATE_STRING(730729)` retorna a sequência "08/31/00".

Nota: Consulte “Macro DATE” na página 51 para obter informações adicionais sobre formatos de data válidos.

Funções relacionadas

Função	Descrição
DATE	Converte uma sequência de datas em uma data Juliana.
DATE_JULIAN	Retorna a data Juliana da entrada.
DATE_FORMAT	Converte datas de um formato para outro.

Macro DAY_BETWEEN

O DAY_BETWEEN está disponível apenas no IBM Campaign.

Sintaxe

```
DAY_BETWEEN(from_date_string, to_date_string [, input_format])
```

Parâmetros

from_date_string

Um texto representando uma data válida a partir da qual contar o número de dias decorridos.

to_date_string

Um texto representando uma data válida até a qual o número de dias é contado. Esta data deve estar no mesmo formato que from_date_string.

input_format

Uma das palavras-chave na tabela abaixo, especificando o formato de data de from_date_string e de to_date_string.

Descrição

DAY_BETWEEN retorna o número de dias entre from_date_string e to_date_string. Se input_format não for fornecido, a palavra-chave padrão DELIM_M_D_Y será utilizada.

Exemplos

DAY_BETWEEN("08/25/00","08/31/00") retorna o número 6.

Nota: Consulte "Macro DATE" na página 51 para obter informações adicionais sobre formatos de data válidos.

Funções relacionadas

Função	Descrição
DAY_FROMNOW	Retorna o número de dias entre o dia atual e uma data especificada.
DAY_INTERVAL	Retorna o número de dias entre duas datas especificadas.

Macro DAY_FROMNOW

O DAY_FROMNOW está disponível apenas no IBM Campaign.

Sintaxe

```
DAY_FROMNOW(to_year, to_month, to_day)
```

Parâmetros

to_year

Número de ano de 2 ou 4 dígitos válido.

to_month

Número do mês válido entre 1-12.

to_day

Número de dia válido entre 1-31.

Descrição

DAY_FROMNOW retorna o número de dias entre o dia atual e a data especificada por to_year/to_month/to_day.

Nota: Se a data especificada estiver no passado, o valor retornado será negativo.

Exemplos

Se a data de hoje for 31 de agosto de 2000, DAY_FROMNOW(2000,12,31) retornará o número 122.

Funções relacionadas

Função	Descrição
DAY_BETWEEN	Retorna o número de dias entre duas sequências de data especificadas.
DAY_INTERVAL	Retorna o número de dias entre duas datas especificadas.

Macro DAY_INTERVAL

O DAY_INTERVAL está disponível apenas no IBM Campaign.

Sintaxe

```
DAY_INTERVAL(from_year, from_month, from_day, to_year, to_month, to_day)
```

Parâmetros

from_year

Número de ano de 2 ou 4 dígitos válido.

from_month

Número do mês válido entre 1-12.

from_day

Número de dia válido entre 1-31.

to_year

Número de ano de 2 ou 4 dígitos válido.

to_month

Número do mês válido entre 1-12.

to_day

Número de dia válido entre 1-31.

Descrição

DAY_INTERVAL retorna o número de dias entre a Data de início especificada (from_year/from_month/from_day) e a Data de conclusão especificada (to_year/to_month/to_day).

Exemplos

DAY_INTERVAL(2000,8,31,2000,12,31) retorna o número 122.

Funções relacionadas

Função	Descrição
DAY_BETWEEN	Retorna o número de dias entre duas sequências de data especificadas.
DAY_FROMNOW	Retorna o número de dias entre o dia atual e uma data especificada.

Macro DAYOF

O DAYOF está disponível apenas no IBM Campaign.

Sintaxe

DAYOF(date_string [, input_format])

Parâmetros

date_string

Um texto representando uma data válida.

input_format

Uma das palavras-chave na tabela abaixo, especificando o formato de data de `date_string`.

Descrição

`DAYOF` retorna o dia do mês como um número para a data representada pelo `date_string`. Se `input_format` não for fornecido, a palavra-chave padrão `DELIM_M_D_Y` será utilizada.

Exemplos

`DAYOF("08/31/00")` retorna o número 31.

Nota: Consulte "Macro DATE" na página 51 para obter informações adicionais sobre formatos de data válidos.

Macro DISTANCE

O `DISTANCE` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
DISTANCE(lat1, long1, lat2, long2[, UNIT_OF_MEASURE][, PRECISION])
```

Parâmetros

`lat1`

A latitude do primeiro ponto, como um valor decimal.

`long1`

A longitude do primeiro ponto, como um valor decimal.

`lat2`

A latitude do segundo ponto, como um valor decimal.

`long2`

A longitude do segundo ponto, como um valor decimal.

`UNIT_OF_MEASURE`

Um parâmetro opcional indicando a unidade de medida de distância retornada. Os valores são MILES ou KILOMETERS. Se omitir esse parâmetro, MILES será o padrão.

`PRECISION`

Um parâmetro opcional que indica o nível de precisão após o ponto decimal para a distância retornada. Se especificar um valor de precisão, a distância retornada será truncada para o número de casas decimais que você especificar. O valor máximo é 5. Se omitir esse valor, o número de casas decimais não será truncado.

Descrição

DISTANCE calcula a distância entre dois pontos. A latitude e a longitude devem estar em unidades decimais. Sempre use uma vírgula e um espaço para separar valores numéricos. Isto é necessário para acomodar idiomas que utilizam uma vírgula como um separador decimal, conforme mostrado no segundo exemplo a seguir.

Exemplos

DISTANCE (18,529747, 73,839798, 18,533511, 73,8777995, MILES, 2) retorna o valor 2,50 Milhas.

DISTANCE (18,529747, 73,839798, 18,533511, 73,8777995, KILOMETERS, 1) retorna o valor de 4,0 quilômetros.

Macro DIV

O DIV está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
data DIV divisor data / divisor
```

Parâmetros

data

Os valores numéricos pelos quais dividir. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

divisor

Um ou mais valores pelos quais dividir os valores no intervalo de dados especificado. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em data2 deve ser igual ao número de colunas em data1, a menos que data2 seja uma constante. Para obter a definição de formato de divisor (mesmo que data), consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

DIV divide o intervalo de dados especificado pelo valor do divisor. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em data1 dividido pela coluna correspondente de data2 (ou seja, a primeira coluna de data1 é dividida pela primeira coluna de data, a segunda coluna com a segunda coluna, e assim por diante).

Se data2 for uma constante, cada valor em data1 será dividido por este valor. Se data2 contiver uma ou mais colunas, os cálculos serão executados em uma base de linha por linha entre uma coluna data2 e uma coluna de data2. A primeira linha de data1 é dividida pelo valor da primeira linha de data2, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: Usar uma coluna contendo o mesmo número *x* em cada linha como `divisor` é o mesmo que utilizar a constante *x* como um `divisor`.

Nota: O operador `DIV` pode ser abreviado com uma barra (/).

Exemplos

`TEMP = 8 DIV 4` ou `TEMP = 8/4` Cria uma nova coluna denominada `TEMP` contendo o valor dois.
`TEMP = V1/8` Cria uma nova coluna denominada `TEMP`, em que cada valor é o conteúdo da coluna `V1` dividido por oito.
`TEMP =V1:V3/2` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são o conteúdo da coluna `V1` dividido por dois, os valores da coluna `VX` são o conteúdo da coluna `V2` dividido por dois, e os valores da coluna `VY` são o conteúdo da coluna `V3` dividida por dois.
`TEMP = V1/V1` Cria uma nova coluna denominada `TEMP` contendo todos os números um (já que qualquer número dividido por ele mesmo é um).
`TEMP = V1/V2` Cria uma nova coluna denominada `TEMP`, em que cada valor é o valor da linha da coluna `V1` dividido pelo valor da linha correspondente da coluna `V2`.
`TEMP = V1:V3/V4:V6` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. A coluna `TEMP` contém os valores em `V1` dividido pelos valores da linha correspondente da coluna `V4`. A coluna `VX` contém a divisão da coluna `V2` por `V5`. A coluna `VY` contém a divisão da coluna `V3` por `V6`.
`TEMP = V1[10:20] / V2` ou `TEMP = V1[10:20] / V2[1:11]` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm o resultado da divisão dos valores nas linhas 10-20 da coluna `V1` pelos valores nas linhas 1-11 da coluna `V2`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`MOD`	Calcula o módulo do conteúdo do intervalo de dados especificado
`MULT`	Multiplica o conteúdo de dois intervalos de dados
`POW`	Calcula o valor base elevado às potências exponenciais especificadas

Macro EQ

O `EQ` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

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

Parâmetros

`data1`

O intervalo de células para comparar. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`data2`

O número ou os números com relação aos quais comparar todos os valores na coluna especificada. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em `data2` deve ser igual ao número de colunas em `data1`, a menos que `data2` seja uma constante. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`EQ` compara os dois intervalos de dados especificados, retornando um se os valores forem iguais ou um zero se não forem iguais. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em `data1` em comparação com a coluna correspondente de `data2` (ou seja, a primeira coluna de `data1` é comparada com a primeira coluna de `data`, a segunda coluna com a segunda coluna, e assim por diante).

Se `data2` for uma constante, cada valor em `data` será comparado com esse valor. Se `data2` for uma coluna, os cálculos serão executados em uma base de linha por linha. Os valores em `data1` são comparados com o valor da primeira linha de `data2`, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Ao comparar sequências, maiúsculas e minúsculas não importam (ou seja, "Yes", "YES", "yes" e "yeS" são todos considerados iguais).

Nota: O operador `EQ` pode ser abreviado com um sinal de igual duplo (`==`). Dentro de parênteses, um sinal de igual único (`=`) também pode ser usado para a função macro `EQ` (fora de parênteses, o sinal de igual é interpretado como o operador de designação).

Exemplos

```
TEMP = 3 EQ 4 ou TEMP = 3==4 ou TEMP = (3=4)
```

Cria uma nova coluna denominada `TEMP` contendo o valor zero (já que três não é igual a quatro).

`TEMP = "No" == "NO"` Cria uma nova coluna denominada `TEMP` contendo o valor um (as comparações de sequências não fazem distinção entre maiúsculas e minúsculas).
`TEMP = V1 == 8` Cria uma nova coluna denominada `TEMP`, em que cada valor é um se o valor da linha correspondente da coluna `V1` for igual ao número oito, caso contrário, zero.
`TEMP = V1==V1` Cria uma nova coluna denominada `TEMP` contendo todos os números um (já que cada número é igual a ele mesmo).
`TEMP = V1==V2` Cria uma nova coluna denominada `TEMP`, em que cada valor é o valor da linha da coluna `V1` em comparação com o valor da linha correspondente da coluna `V2`.
`TEMP = V1:V3 == V4:V6` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. A coluna `TEMP` contém os valores em `V1` em comparação com os valores da linha correspondente da coluna `V4`. A coluna `VX` compara as colunas `V2` e `V5`. A coluna `VY` compara as colunas `V3` e `V6`.
`TEMP = V1[10:20] == V2` ou `TEMP = V1[10:20] == V2[1:11]` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os resultados da comparação dos valores nas linhas 10-20 da coluna `V1` com as linhas 1-11 da coluna `V2`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`EQ`	Retornará TRUE se um intervalo de dados for igual ao outro
`GE`	Retornará TRUE se um intervalo de dados for maior ou igual ao outro
`GT`	Retornará TRUE se um intervalo de dados for maior que o outro
`LE`	Retornará TRUE se um intervalo de dados for menor ou igual ao outro
`LT`	Retornará TRUE se um intervalo de dados for menor que o outro
`NE`	Retornará TRUE se um intervalo de dados não for igual ao outro

Macro EXP

O `EXP` está disponível apenas no IBM Campaign.

Sintaxe

```
EXP(data)
```

Parâmetros

`data`

Os valores numéricos utilizados como um expoente para o número natural, *e*. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

O `EXP` eleva o número natural, *e*, por cada um dos valores no intervalo de dados especificado (ou seja, calcula e^x). A constante *e* é igual a 2,7182818. O `EXP` retorna uma nova coluna para cada coluna de entrada, cada uma contendo o resultado do e^x para cada valor de *x* na coluna ou colunas de entrada correspondentes. O `EXP` é o inverso da função macro `LN`.

Nota: Se o valor de *x* for muito grande ou muito pequeno, um erro de estouro será retornado. Isso ocorrerá se e^x exceder o valor de vírgula flutuante máximo ou mínimo de 32 bits.

Exemplos

`TEMP = EXP(2)` Cria uma nova coluna denominada `TEMP` contendo o valor `7.39`.
`TEMP = EXP(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é resultado de elevar o *e* ao conteúdo da coluna `V1`.
`TEMP = EXP(V1:V3)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os resultados de elevar o *e* à coluna `V1`, os valores da coluna `VX` são os resultados de elevar o *e* ao conteúdo da coluna `V2`, e os valores da coluna `VY` são os resultados de elevar o *e* ao conteúdo da coluna `V3`.
`TEMP = EXP(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os resultados de elevar o *e* aos valores nas linhas 10-20 da coluna `V1`. Outras células em `TEMP` estão vazias.
`TEMP = EXP(V1[1:5]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são os resultados de elevar o *e* aos valores da linha correspondente da coluna `V1`, e os valores na coluna `VX` são os resultados de elevar o *e* aos valores da linha correspondente da coluna `V2`.

Funções relacionadas

Função	Descrição
`LN` ou `LOG`	Calcula o log natural do conteúdo do intervalo de dados especificado
`LOG2`	Calcula o log de base2 do conteúdo do intervalo de dados especificado
`LOG10`	Calcula o log de base10 do conteúdo do intervalo de dados especificado

Função	Descrição
POW	Potência exponencial

Macro EXTERNALCALLOUT

A macro EXTERNALCALLOUT está disponível apenas no IBM Interact.

Sintaxe

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

Parâmetros

calloutName

O nome do texto explicativo criado usando a API ExternalCallout. Esse nome deve corresponder ao nome da categoria ExternalCallout que você criou no IBM Marketing Platform.

arg1

Um argumento requerido por seu texto explicativo, se necessário.

Descrição

EXTERNALCALLOUT permite chamar um aplicativo externo para incluir dados em seu fluxograma interativo. O EXTERNALCALLOUT pode retornar tudo o que você tiver criado para o texto explicativo executar. Deve-se gravar esse texto explicativo em Java usando a API ExternalCallout. Para obter mais detalhes, consulte o *Guia do Desenvolvedor do IBM Interact*.

Exemplos

EXTERNALCALLOUT(getStockPrice, UNCA)

Chama o texto explicativo getStockPrice transmitindo o nome da ação, UNCA, como argumento. Este texto explicativo definido pelo usuário retorna o preço da ação, conforme definido pelo texto explicativo.

Macro FACTORIAL

O FACTORIAL está disponível apenas no IBM Campaign.

Sintaxe

FACTORIAL(data)

Parâmetros

data

Os valores de números inteiros a partir dos quais calcular o fatorial. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima, mas deve ser maior ou igual a zero. Para obter

a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

FACTORIAL calcula o fatorial dos valores no intervalo de dados especificado. Todas as entradas devem ser números inteiros maiores ou iguais a zero. O fatorial de um número inteiro menor ou igual a um é um. Para números inteiros X ≥ 2, o fatorial $x! = x\ (x\text{-}1)(x\text{-}2)...(x\ -\ (x\text{-}1))$. FACTORIAL retorna uma nova coluna para cada coluna de entrada, cada uma contendo o fatorial dos números na coluna de entrada correspondente.

Nota: Qualquer valor maior que 34 produzirá ??? (erro no estouro da vírgula flutuante).

Exemplos

TEMP = FACTORIAL(3) Cria uma nova coluna denominada TEMP contendo o valor 6.
TEMP = FACTORIAL(-2) Gera um erro 333, indicando que o argumento deve ser maior ou igual a 0.
TEMP = FACTORIAL(V1) Cria uma nova coluna denominada TEMP, em que cada valor é o fatorial do conteúdo da coluna V1.
TEMP = FACTORIAL(V1:V3) Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os fatoriais do conteúdo da coluna V1, os valores da coluna VX são os fatoriais do conteúdo da coluna V2, e os valores da coluna VY são os fatoriais do conteúdo da coluna V3.
TEMP = FACTORIAL(V1[10:20]) Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm os fatoriais dos valores nas linhas 10-20 da coluna V1. Outras células em TEMP estão vazias.
TEMP = FACTORIAL(V1[50:99]:V2) Cria duas novas colunas denominadas TEMP e VX, cada uma com valores nas linhas 1-50 (as outras células estão vazias). Os valores na coluna TEMP são os fatoriais das linhas da coluna V1, e os valores na coluna VX são os fatoriais dos valores na coluna V2.

Macro FLOOR

O FLOOR está disponível apenas no IBM Campaign.

Sintaxe

FLOOR(data)

Parâmetros

data

Os valores numéricos dos quais calcular o piso. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer

item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

FLOOR calcula o piso dos valores no intervalo de dados especificado. O piso de um número é o maior número inteiro, que é menor que o número. FLOOR retorna uma nova coluna para cada coluna de entrada, cada uma contendo o piso dos números na coluna de entrada correspondente.

Nota: Isto é o mesmo que a função macro INT.

Exemplos

TEMP = FLOOR(4.3) Cria uma nova coluna denominada TEMP contendo o valor 4.
TEMP = FLOOR(2.9) Cria uma nova coluna denominada TEMP contendo o valor -3.
TEMP = FLOOR(V1) Cria uma nova coluna denominada TEMP, em que cada valor é o piso do conteúdo da coluna V1.
TEMP = FLOOR(V1:V3) Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os pisos do conteúdo da coluna V1, os valores da coluna VX são os pisos do conteúdo da coluna V2, e os valores da coluna VY são os pisos do conteúdo da coluna V3.
TEMP = FLOOR(V1[10:20]) Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm os pisos dos valores nas linhas 10-20 da coluna V1. Outras células em TEMP estão vazias.
TEMP = FLOOR(V1[50:99]:V2) Cria duas novas colunas denominadas TEMP e VX, cada uma com valores nas linhas 1-50 (as outras células estão vazias). Os valores na coluna TEMP são os pisos das linhas da coluna V1, e os valores na coluna VX são os pisos dos valores na coluna V2.

Funções relacionadas

Função	Descrição
CEILING	Calcula o limite de cada valor no intervalo de dados especificado
FRACTION	Retorna a parte fracionária de cada valor no intervalo de dados especificado.
TRUNCATE	Retorna a parte não fracionária de cada valor no intervalo de dados especificado.

Macro FORMAT

O FORMAT está disponível apenas no IBM Campaign.

Sintaxe

Format possui duas formas, uma para tipos de dados numéricos e outra para tipos de dados de texto/caracteres.

Para tipos de dados numéricos:

```
FORMAT(colName, width [, precision [, format_type [, alignment [,
padding]]]])
```

Para tipos de dados de texto/caractere:

```
FORMAT(colName, width [, alignment])
```

Parâmetros

`colName`

A macro examina o colName, determina seu tipo de dados e, em seguida, impõe as regras apropriadas para parâmetros subsequentes adequadamente.

`width`

A largura deve ser grande o suficiente para conter o resultado completo, caso contrário, o resultado será truncado. Os valores aceitáveis vão de 1 a 29, se colName for numérico, caso contrário, de 1 a 255.

`precision`

Precisão é o número de dígitos após o ponto decimal. Os valores aceitáveis vão de 0 a 15. Se for zero, então o resultado será um número inteiro. O valor de precisão padrão é 2.

`format_type`

As palavras-chave válidas para format_type são:

PERIOD	Ponto(.) é usado como símbolo decimal. Nenhum símbolo de agrupamento por dígito é usado. Este é o valor padrão.
COMMA	Vírgula(,) é usada como símbolo decimal. Nenhum símbolo de agrupamento por dígito é usado.
PERIOD_COMMA	Ponto como símbolo decimal e vírgula como símbolo de agrupamento por dígito.
COMMA_PERIOD	Vírgula como símbolo decimal e ponto como símbolo de agrupamento por dígito.

`alignment`

As palavras-chave válidas para alinhamento são LEFT e RIGHT. O valor padrão é RIGHT para tipos de dados numéricos e LEFT para tipos de dados de texto/caracteres.

`padding`

As palavras-chave válidas para preenchimento são SPACE e ZERO. O valor padrão é SPACE. ZERO será ignorado (e SPACE utilizado) se o alinhamento for LEFT.

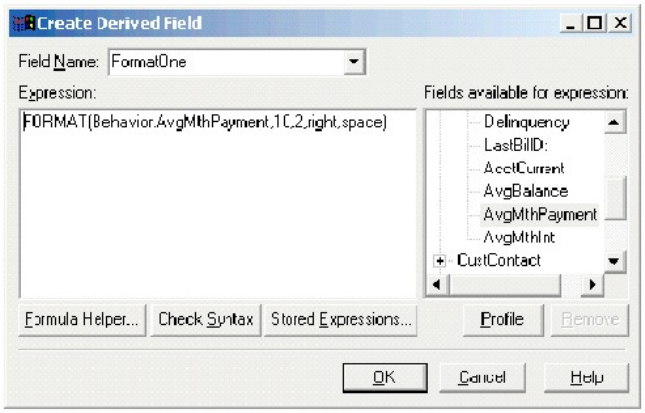
Observe que as sequências numéricas mantidas em um tipo de dados de texto/caracteres são tratadas como texto/caracteres. Além disso, observe que a forma numérica usa diversas palavras-chave opcionais, cada uma com um valor padrão. Entretanto, para substituir o padrão da segunda palavra-chave opcional ou subsequentes, DEVE-SE codificar os padrões para as palavras-chave opcionais anteriores (como resultado, eles se tornam obrigatórios). Por exemplo: para substituir o alinhamento para que seja LEFT, deve-se codificar: `FORMAT(myNumCol, 10, 2, PERIOD, LEFT)`.

Descrição

`FORMAT` converte dados numéricos em uma forma de sequência com várias opções de formatação para controlar e definir a sequência de saída. Isso será especialmente útil para criação de arquivos de captura instantânea com formatos específicos para fins de arquivo de correspondência.

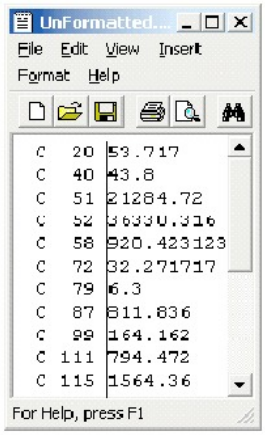
Exemplos

O exemplo a seguir define um campo derivado usando `FORMAT`.

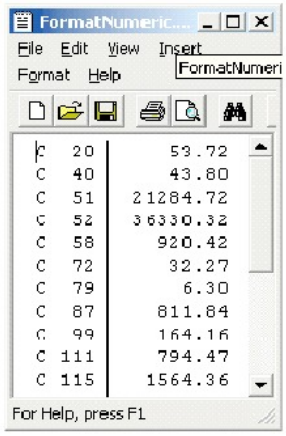


Os exemplos a seguir mostram o mesmo campo, `AvgMthPayment`, em três formatos.

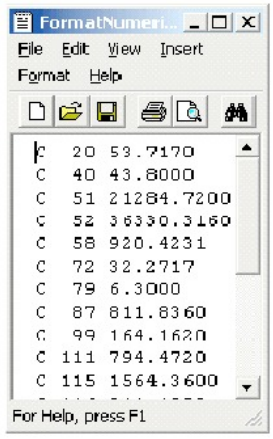
Não-formatado:



Formatado usando `FORMAT(Behavior.AvgMthPayment,10,2,right,space)`:



Formatado usando `FORMAT(Behavior.AvgMthPayment,10,4)`:



Macro FRAÇÃO

O `FRACTION` está disponível apenas no IBM Campaign.

Sintaxe

```
FRACTION(data)
```

Parâmetros

`data`

Os valores numéricos dos quais calcular a fração. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`FRACTION` calcula a parte fracionária dos valores no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a parte fracionária dos números na coluna de entrada correspondente.

Nota: As funções macro `FRACTION` e `TRUNCATE` são complementares na medida em que elas somam para os valores originais.

Exemplos

`TEMP = FRACTION(4.3)` Cria uma nova coluna denominada `TEMP` contendo o valor `0.3`.
`TEMP = FRACTION(2.9)` Cria uma nova coluna denominada `TEMP` contendo o valor `-0.9`.
`TEMP = FRACTION(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é a parte fracionária do conteúdo da coluna `V1`.
`TEMP = FRACTION(V1:V3)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são as partes fracionárias do conteúdo da coluna `V1`, os valores da coluna `VX` são as partes fracionárias do conteúdo da coluna `V2`, e os valores da coluna `VY` são as partes fracionárias do conteúdo da coluna `V3`.
`TEMP = FRACTION(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm as partes fracionárias dos valores nas linhas 10-20 da coluna `V1`. Outras células em `TEMP` estão vazias.
TEMP = FRACTION(V1[50:99]:V2) Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-50 (as outras células estão vazias). Os valores na coluna `TEMP` são as partes fracionárias das linhas da coluna `V1`, e os valores na coluna `VX` são as partes fracionárias dos valores na coluna `V2`.

Funções relacionadas

Função	Descrição
`CEILING`	Calcula o limite de cada valor no intervalo de dados especificado

Função	Descrição
FLOOR	Calcula o piso de cada valor no intervalo de dados especificado.
TRUNCATE	Retorna a parte não fracionária de cada valor no intervalo de dados especificado.

Macro GE

A macro GE está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
data1 GE data2 data1 >= data2
```

Parâmetros

`data1`

O intervalo de células numéricas para comparar. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`data2`

O número ou os números com relação aos quais comparar todos os valores na coluna especificada. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em `data2` deve ser igual ao número de colunas em `data1`, a menos que `data2` seja uma constante. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`GE` compara os dois intervalos de dados especificados, retornando um se os valores no primeiro conjunto de dados forem maiores ou iguais aos valores no segundo conjunto de dados ou, caso contrário, um zero. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em `data1` em comparação com a coluna correspondente de `data2` (ou seja, a primeira coluna de `data1` é comparada com a primeira coluna de `data`, a segunda coluna com a segunda coluna, e assim por diante).

Se `data2` for uma constante, cada valor em `data` será comparado com esse valor. Se `data2` for uma coluna, os cálculos serão executados em uma base de linha por linha. Os valores em `data1` são comparados com o valor da primeira linha de `data2`, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: O operador `GE` pode ser abreviado com um sinal de maior que seguido por um sinal de igual (`>=`).

Exemplos

`TEMP = 9 GE 4` ou `TEMP = 9 >= 4` Cria uma nova coluna denominada `TEMP` contendo o valor um (já que nove é maior que quatro).
`TEMP = V1 >= 8` Cria uma nova coluna denominada `TEMP`, em que cada valor será um se o valor da linha correspondente da coluna `V1` for maior ou igual ao número oito, caso contrário, zero.
`TEMP = V1:V3 >= 2` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são o conteúdo da coluna `V1` em comparação com o valor dois, os valores da coluna `VX` são o conteúdo da coluna `V2` em comparação com o valor dois, e os valores da coluna `VY` são o conteúdo da coluna `V3` em comparação com o valor dois.
`TEMP = V1 >= V1` Cria uma nova coluna denominada `TEMP` contendo todos os números um (já que cada número é igual a ele mesmo).
`TEMP = V1 >= V2` Cria uma nova coluna denominada `TEMP`, em que cada valor é o valor da linha da coluna `V1` em comparação com o valor da linha correspondente da coluna `V2`.
`TEMP = V1:V3 >= V4:V6` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. A coluna `TEMP` contém os valores em `V1` em comparação com os valores da linha correspondente da coluna `V4`. A coluna `VX` compara as colunas `V2` e `V5`. A coluna `VY` compara as colunas `V3` e `V6`.
`TEMP = V1[10:20] >= V2` ou `TEMP = V1[10:20] >= V2[1:11]` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os resultados da comparação dos valores nas linhas 10-20 da coluna `V1` com os valores nas linhas 1-11 da coluna `V2`. Outras células em `TEMP` estão vazias.

Funções relacionadas

`NE`	Retornará TRUE se um intervalo de dados não for igual ao outro

Macro GROUPBY

O `GROUPBY` está disponível apenas no IBM Campaign.

Sintaxe

```
GROUPBY(group_field, keyword, rolled_field [,output_field])
```

Parâmetros

- `group_field`

  Especifica a variável sobre quais registros serão agrupados (isto é, todos os mesmos valores da variável especificada são agrupados).

- `keyword`

  Especifica a função de acumulação de resumo a ser executada para o campo acumulado.

- `rolled_field`

  Especifica a variável a ser resumida ou acumulada.
- `output_field`

  Identifica uma variável alternativa a ser retornada para uma única linha de um grupo e pode ser usada apenas com as palavras-chave `MinOf`, `MaxOf` e `MedianOf`.

Descrição

`GROUPBY` resume entre diversas linhas de dados em um grupo. A saída desta função é uma coluna única. A saída é o resultado da operação especificada pela `keyword` no `rolled_field` ao longo do grupo homogêneo especificado pelo `group_field`. Se houver mais de uma resposta que satisfaça uma condição especificada, a primeira localizada será retornada.

Se o `output_field` opcional não for fornecido, então o resultado será a saída da operação no `rolled_field`. Se `output_field` for fornecido, o resultado será o `output_field` da linha dentro do grupo.

Se houver diversas linhas dentro de um grupo que satisfaçam a condição especificada (por exemplo, existem laços para o valor máximo), o `output-field` associado com a primeira linha que satisfizer a condição será retornado.

Nota: Para trabalhar com agrupamento em várias colunas, é possível colocar uma lista de nomes de campos, separados por vírgulas, dentro de um conjunto de chaves "{ }" e utilizar isso como o primeiro parâmetro na denominada de macro `GROUPBY`.

As palavras-chave suportadas são as seguintes (sem distinção entre maiúsculas e minúsculas):

Palavra-chave	**Sequência? Sim/Não**	**Descrição**
`CountOf`	`Yes`	Retorna o número de registros em cada grupo (`rolled_field` pode ser número ou sequência; o valor retornado é o mesmo, independentemente do valor de `rolled_field`).
`MinOf`	`Yes`	Retorna o valor mínimo de `rolled_field` em cada grupo (`rolled_field` pode ser um número ou uma sequência; se `rolled_field` for uma sequência, o valor mais próximo ao início do alfabeto em que foi classificado em ordem alfabética será retornado).

Palavra-chave	Sequência? Sim/Não	Descrição
MaxOf	Yes	Retorna o valor máximo de rolled_field em cada grupo (rolled_field pode ser um número ou uma sequência; se rolled_field for uma sequência, o valor mais próximo ao início do alfabeto em que foi classificado em ordem alfabética será retornado).
DiffOf	Yes	Retorna o número de valores distintos de rolled_field em cada grupo (rolled_field pode ser número ou uma sequência).
AvgOf	No	Retorna o valor médio de rolled_field em cada grupo (rolled_field deve ser numérico).
ModeOf	Yes	Retorna o valor modal (ou seja, o valor que ocorrer com mais frequência) de rolled_field em cada grupo (rolled_field pode ser um número ou uma sequência).
MedianOf	Yes	Retorna o valor mediano (ou seja, o valor médio quando classificado por rolled_field) de rolled_field em cada grupo (rolled_field pode ser numérico ou uma sequência; se rolled_field for uma sequência, os valores serão classificados em ordem alfabética).
OrderOf	Yes	Retorna a ordem de rolled_field em cada grupo (rolled_field deve ser numérico). Se diversos registros tiverem o mesmo valor, todos eles receberão o mesmo valor.
SumOf	No	Retorna a soma de rolled_field em cada grupo (rolled_field deve ser numérico).
StdevOf	No	Retorna o desvio padrão de rolled_field em cada grupo (rolled_field deve ser numérico).

Palavra-chave	Sequência? Sim/Não	Descrição
IndexOf	Yes	Retorna o índice de base1 (ordenado pelo rolled_field) de cada registro (rolled_field pode ser número ou uma sequência). A ordem de classificação é crescente. Nota: Para campos numéricos, as ordens de classificação RankOf e IndexOf podem se tornar decrescentes ao colocar um sinal de menos (-) em frente ao campo de classificação.
RankOf	Yes	Retorna a categoria de base 1 (ordenada por rolled_field) na qual cada registro se encontra (rolled_field pode ser número ou uma sequência). A ordem de classificação é crescente. Nota: Para campos numéricos, as ordens de classificação RankOf e IndexOf podem se tornar decrescentes ao colocar um sinal de menos (-) em frente ao campo de classificação.

Exemplos

`GROUPBY (Household_ID, SumOf, Account_Balance)` Calcula a soma de todos os saldos de contas por família.
`GROUPBY (Cust_ID, MinOf, Date(Account_Open_Date), Acc_Num)` Retorna o número da conta da primeira conta aberta pelo cliente.

macro GROUPBY_WHERE

O GROUPBY_WHERE está disponível apenas no IBM Campaign.

Sintaxe

```
GROUPBY_WHERE(group_field, keyword, rolled_field, where_value
[,output_field])
```

Parâmetros

- group_field

  Especifica a variável sobre quais registros serão agrupados (isto é, todos os mesmos valores da variável especificada são agrupados).

- keyword

  Especifica a função de acumulação de resumo a ser executada.
- rolled_field

  Especifica a variável a ser resumida ou acumulada.
- where_value

  Uma expressão que é avaliada para um valor um ou zero que especifica quais linhas devem ser incluídas na operação de acumulação.
- output_field

  Identifica uma variável alternativa a ser retornada para uma única linha de um grupo e pode ser usada apenas com as palavras-chave MinOf, MaxOf e MedianOf

Descrição

GROUPBY_WHERE resume entre as linhas específicas de dados em um grupo. A saída desta função é uma coluna única. A saída é o resultado da operação especificada pela keyword no rolled_field ao longo do grupo homogêneo especificado pelo group_field, filtrada pelo where_value. Apenas as linhas com um where_value de um são incluídas no cálculo.

Se o output_field opcional não for fornecido, então o resultado será o resultado da operação no rolled_field. Se output_field for fornecido, o resultado será o output_field da linha dentro do grupo.

Nota: Consulte "Macro GROUPBY" na página 73 para obter mais informações sobre os valores válidos para keyword.

Exemplos

GROUPBY_WHERE (Household_ID, SumOf, Account_Balance, Account_Balance>0) Calcula a soma de todas as contas com saldos positivos para cada família.
GROUPBY_WHERE (Cust_ID, AvgOf, Purchase_Amt,Date(Current_Date) - Date(Purchase_Date)<90) Calcula o valor médio de compra de cada cliente nos últimos 90 dias.

Macro GT

O GT está disponível no IBM Campaign e no IBM Interact.

Sintaxe

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

Parâmetros

data1

O intervalo de células numéricas para comparar. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

data2

Os números com relação aos quais comparar todos os valores na coluna especificada. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em data2 deve ser igual ao número de colunas em data1, a menos que data2 seja uma constante. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

GT compara os dois intervalos de dados especificados, retornando um se os valores no primeiro conjunto de dados forem maiores que os valores no segundo conjunto de dados ou, caso contrário, um zero. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em data1 em comparação com a coluna correspondente de data2 (ou seja, a primeira coluna de data1 é comparada com a primeira coluna de data, a segunda coluna com a segunda coluna, e assim por diante).

Se data2 for uma constante, cada valor em data será comparado com esse valor. Se data2 for uma coluna, os cálculos serão executados em uma base de linha por linha. Os valores em data1 são comparados com o valor da primeira linha de data2, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: O operador GT pode ser abreviado com um sinal de maior que (>).

Exemplos

TEMP = 3 GT 4 ou TEMP = 3 > 4 Cria uma nova coluna denominada TEMP contendo o valor zero (já que três não é maior que quatro).
TEMP = V1 > 8 Cria uma nova coluna denominada TEMP, em que cada valor será um se o valor da linha correspondente da coluna V1 for maior que o número oito, caso contrário, zero.
TEMP = V1:V3 > 2 Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são o conteúdo da coluna V1 em comparação com o valor dois, os valores da coluna VX são o conteúdo da coluna V2 em comparação com o valor dois, e os valores da coluna VY são o conteúdo da coluna V3 em comparação com o valor dois.
TEMP = V1 > V1 Cria uma nova coluna denominada TEMP contendo todos os zeros (já que nenhum número é maior que ele mesmo).
TEMP = V1 > V2 Cria uma nova coluna denominada TEMP, em que cada valor é o valor da linha da coluna V1 em comparação com o valor da linha correspondente da coluna V2.
TEMP = V1:V3 > V4:V6 Cria três novas colunas denominadas TEMP, VX e VY. A coluna TEMP contém os valores em V1 em comparação com os valores da linha correspondente da coluna V4. A coluna VX compara as colunas V2 e V5. A coluna VY compara as colunas V3 e V6.

`TEMP = V1[10:20] > V2` ou `TEMP = V1[10:20] > V2[1:11]` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os resultados da comparação dos valores nas linhas 10-20 da coluna `V1` com os valores nas linhas 1-11 da coluna `V2`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`EQ`	Retornará TRUE se um intervalo de dados for igual ao outro
`GE`	Retornará TRUE se um intervalo de dados for maior ou igual ao outro
`LE`	Retornará TRUE se um intervalo de dados for menor ou igual ao outro
`LT`	Retornará TRUE se um intervalo de dados for menor que o outro
`NE`	Retornará TRUE se um intervalo de dados não for igual ao outro

Macro IF

O `IF` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
IF(predicate_col, then_value) IF(predicate_col, then_value, else_value)
```

Parâmetros

`predicate_col`

Uma coluna de valores booleanos ou uma expressão avaliada para uma coluna única de valores booleanos. Os valores booleanos são interpretados como zero ou diferente de zero. Esta coluna deve conter pelo menos tantas linhas quanto o intervalo de dados do qual os dados estão sendo extraídos.

`then_value`

Um ou mais valores para retornar se a linha correspondente de `predicate_col` contiver um valor diferente de zero. Isso pode ser um valor constante, uma coluna ou uma expressão avaliada para qualquer item acima. Consulte “Parâmetros da função macro para o IBM Campaign” na página 9 para obter a definição de formato de `then_value` (o mesmo que `data`).

`else_value`

Se esse parâmetro opcional for fornecido, ele será retornado se a linha correspondente de `predicate_col` contiver um zero. Isso pode ser um valor constante, uma coluna ou uma expressão avaliada para qualquer item acima. Se `else_value` não for fornecido, um zero será retornado sempre que `predicate_col` for avaliado para false. Consulte “Parâmetros da função macro para o IBM

Campaign" na página 9 para obter a definição de formato de else_value (o mesmo que data).

Descrição

IF avalia a expressão em predicate_col e retorna then_value se a expressão for true ou else_value se a expressão for false. Ele retorna o mesmo número de colunas em then_value e else_value. Uma ou mais novas colunas conterão um ou mais valores correspondentes a partir de then_value se o valor de predicate_col for diferente de zero. Se else_value for fornecido, ele será retornado quando o valor de predicate_col for zero. Se else_value não for fornecido, zero será retornado.

Como IF opera em uma base de linha por linha, ele produz um resultado para cada linha até o último valor da coluna mais curta (ou seja, a coluna mais curta fora de predicate_col, then_value e else_value).

Nota: Geralmente, você desejará criar uma coluna de predicado utilizando uma das funções macro de comparação (por exemplo, ==, >, <, ISEVEN, ISODD, e assim por diante).

Exemplos

TEMP = IF(1, V1)Cria uma nova coluna denominada TEMP contendo uma cópia da coluna V1.
TEMP = IF(V1, 1, 0) Cria uma nova coluna denominada TEMP, em que cada valor será um se o valor correspondente da coluna V1 for diferente de zero, caso contrário, será zero.
TEMP = IF(V3, V1, V2) Cria uma nova coluna denominada TEMP, em que cada valor é copiado a partir da coluna V1 se o valor correspondente da coluna V3 for diferente de zero; caso contrário, o valor será copiado da coluna V2.
TEMP = IF(ABS(V1-AVG(V1)) < STDV(V1), V1) Cria uma nova coluna denominada TEMP contendo cada valor na coluna V1 que for menor que um desvio padrão para fora da média.
TEMP = IF(V3[20:30], V1[30:40], V2) Cria uma nova coluna denominada TEMP contendo valores para as linhas 10-20. Cada valor é copiado a partir da coluna V1 (células 10-20) se o valor correspondente da coluna V3 (células 30-40) for diferente de zero; caso contrário, o valor será copiado da coluna V2 (células 1-11).

Macro IN

O IN está disponível no IBM Campaign e no IBM Interact.

Sintaxe

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

Parâmetros

A primeira forma permite utilizar uma lista de valores ao invés de uma subconsulta.

A segunda forma usa uma subconsulta que é avaliada para produzir um resultado intermediário, com relação ao qual um processamento adicional pode ser executado.

Descrição

O predicado IN permite utilizar uma lista de valores ao invés de uma subconsulta, ou introduzirá uma subconsulta.

Nota: O predicado IN possui uma versão negativa, NOT IN. O formato para isso é idêntico a IN. NOT IN será true apenas se o valor fornecido não for localizado nos valores retornados pela subconsulta.

Importante: Ao usar `IN` no IBM Interact, é possível utilizar apenas a sintaxe `IN (value IN (value1 AND value2 . . . .)`.

Exemplos

`TEMP = IN(25, COLUMN(1...10))` Retorna uma ou mais colunas especificadas de um intervalo de dados
`TEMP = IN("cat", COLUMN("cat", "dog", "bird"))` Cria uma nova coluna denominada `TEMP` contendo o valor um.
`TEMP = IN(V1, V1)` Cria uma nova coluna denominada `TEMP` contendo todos os números um.
`TEMP = IN(V1, V2)` Cria uma nova coluna denominada `TEMP`, em que cada valor será um se a linha correspondente da coluna `V1` contiver um valor na coluna `V2`, mais um zero.

Macro INT

O `INT` está disponível apenas no IBM Campaign.

Sintaxe

`INT(data)`

Parâmetros

`data`

Os valores numéricos para arredondar para baixo para um valor de número inteiro. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

INT calcula o maior número inteiro que é menor que os valores (também conhecido como o piso) no intervalo de dados especificado. INT retorna uma nova coluna para cada coluna de entrada, cada uma contendo o piso de números na coluna de entrada correspondente.

Nota: Isto é o mesmo que a função macro FLOOR.

Exemplos

TEMP = INT(4.7) Cria uma nova coluna denominada TEMP contendo o valor 4.
TEMP = INT(-1.5) Cria uma nova coluna denominada TEMP contendo o valor -2.
TEMP = INT(V1) Cria uma nova coluna denominada TEMP, em que cada valor é o número inteiro maior que é menor ou igual ao conteúdo da coluna V1.
TEMP = V1 - INT(V1) Cria uma nova coluna denominada TEMP que contém a parte decimal de cada valor na coluna V1.
TEMP = INT(V1:V3) Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os números inteiros maiores, que são menores ou iguais ao conteúdo da coluna V1, os valores da coluna VX são os números inteiros maiores, que são menores ou iguais ao conteúdo da coluna V2, e os valores da coluna VY são os números inteiros maiores, que são menores ou iguais ao conteúdo da coluna V3.
TEMP = INT(V1[10:20]) Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm os números inteiros maiores, que são menores ou iguais aos valores correspondentes nas linhas 10-20 da coluna V1. Outras células em TEMP estão vazias.
TEMP = INT(V1[1:5]:V2) Cria duas novas colunas denominadas TEMP e VX, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna TEMP são os números inteiros maiores, que são menores ou iguais aos valores da linha correspondente da coluna V1, e os valores na coluna VX são os números inteiros maiores, que são menores ou iguais aos valores da linha correspondente da coluna V2.

Funções relacionadas

Função	Descrição
ROUND	Calcula o valor arredondado do conteúdo do intervalo de dados especificado
TRUNCATE	Retorna a parte não fracionária de cada valor no intervalo de dados especificado.

Macro INVERSE

O INVERSE está disponível apenas no IBM Campaign.

Sintaxe

```
INVERSE(data)
```

Parâmetros

`data`

Os valores numéricos dos quais calcular o inverso. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`INVERSE` calcula o negativo dos valores no intervalo de dados especificado. Ele retorna *-x* (ou seja, valores negativos são retornados como valores positivos, e valores positivos são retornados como valores negativos). `INVERSE` retorna uma nova coluna para cada coluna de entrada, cada uma contendo o inverso dos valores na coluna de entrada correspondente.

Nota: Para inverter um valor ou uma coluna, preceda-o com um sinal de menos (–). Por exemplo, `V2 = -V1` é o mesmo que `V2 = INVERSE(V1)`.

Exemplos

`TEMP = INVERSE(3.2)` Cria uma nova coluna denominada `TEMP` contendo o valor `-3.2`.
`TEMP = INVERSE(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é o negativo dos valores na coluna `V1`.
`TEMP = INVERSE(V1:V3)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os negativos dos valores na coluna `V1`, os valores da coluna `VX` são os negativos dos valores na coluna `V2`, e os valores da coluna `VY` são os negativos dos valores na coluna `V3`.
`TEMP = INVERSE(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os negativos dos valores nas linhas 10-20 da coluna `V1`. Outras células em `TEMP` estão vazias.
`TEMP = INVERSE(V1[1:5]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são os negativos dos valores das linhas correspondentes da coluna `V1`, e os valores na coluna `VX` são os negativos dos valores das linhas correspondentes da coluna `V2`.

Funções relacionadas

Função	Descrição
ABS	Calcula o valor absoluto do conteúdo do intervalo de dados especificado
NOT	Calcula o NOT lógico do conteúdo do intervalo de dados especificado
SIGN	Calcula o sinal (positivo ou negativo) dos valores no intervalo de dados especificado

Macro IS

O IS está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
IS <keyword>
```

Parâmetros

keyword

Condição de procura, geralmente "NULL," "TRUE," "UNKNOWN," e "FALSE."

Descrição

IS é utilizado em condições de procura complexas. Quanto mais complexa for a procura, mais útil a condição IS poderá ser. Essas condições de procura booleana fornecem uma forma alternativa de expressar as condições de procura básicas.

IS retorna resultados diferentes no IBM Interact a partir do IBM Campaign. NULL retornará 1 se houver pelo menos um valor NULL para um ID do público. UNKNOWN retorna 1 para um ID público se ele não tiver nenhum valor.

Macro ISERROR

O ISERROR está disponível apenas no IBM Campaign.

Sintaxe

```
ISERROR(data)
```

Parâmetros

data

Os valores para testar se alguma das linhas contém um erro (ou seja, uma célula ???). Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

ISERROR verifica se alguma célula de cada linha do intervalo de dados especificado contém um erro (ou seja, uma célula ???). Ele retorna uma nova coluna, com cada linha contendo um valor um se a linha correspondente de data contiver um erro. Caso contrário, ela conterá um zero. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais longa.

Exemplos

TEMP = ISERROR(-3) Cria uma nova coluna denominada TEMP contendo o valor zero.
TEMP = ISERROR(V1) Cria uma nova coluna denominada TEMP, em que cada valor será um se a linha correspondente da coluna V1 contiver ???, caso contrário, um zero.
TEMP = ISERROR(V1:V3) Cria uma nova coluna denominada TEMP, em que cada valor será um se alguma das células na linhas correspondentes da coluna V1 – V3 contiver ???, caso contrário, um zero.
TEMP = ISERROR(V1[50:100]:V10) Cria uma nova coluna denominada TEMP, com valores nas linhas 1-50. Cada valor será um se alguma das células nas linhas 50-100 de colunas V1 – V10 contiver ???, caso contrário, um zero.

Macro ISODD

O ISODD está disponível apenas no IBM Campaign.

Sintaxe

```
ISODD(data)
```

Parâmetros

data

Os valores numéricos para testar se eles são ímpares. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

ISODD testa se cada valor no conjunto de dados especificado é ímpar. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo um para todos os valores ímpares (ou seja, o módulo de valor dois é um) ou um zero para todos os valores não ímpares (isto é, valores pares).

Nota: Para valores de número não inteiro, a função macro INT é aplicada primeiro. Por exemplo, ISODD(2.5) = 0, já que 2 não é ímpar.

Exemplos

`TEMP = ISODD(-3)` Cria uma nova coluna denominada `TEMP` contendo o valor um.
`TEMP = ISODD(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é o resultado do teste do conteúdo da coluna `V1` para detectar valores ímpares.
`TEMP = ISODD(V1:V3)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os resultados do teste do conteúdo da coluna `V1` para detectar valores ímpares, os valores da coluna `VX` são os resultados do teste do conteúdo da coluna `V2` para detectar valores ímpares, e os valores da coluna `VY` são os resultados do teste do conteúdo da coluna `V3` para detectar valores ímpares.
`TEMP = ISODD(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os resultados dos testes dos valores nas linhas 10-20 da coluna `V1` para detectar valores ímpares. Outras células em `TEMP` estão vazias.
`TEMP = ISODD(V1[1:5]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são os resultados do teste das linhas correspondentes da coluna `V1` para detectar valores ímpares, e os valores na coluna `VX` são os resultados do teste das linhas correspondentes da coluna `V2` para detectar valores ímpares.

Funções relacionadas

Função	Descrição
`ISEVEN`	Testa se os valores de entrada são pares (ou seja, divisíveis por dois)

Macro ISEVEN

O `ISEVEN` está disponível apenas no IBM Campaign.

Sintaxe

```
ISEVEN(data)
```

Parâmetros

`data`

Os valores numéricos para testar se eles são pares. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`ISEVEN` testa cada valor no conjunto de dados especificado para detectar valores pares. Ele retorna uma nova coluna para cada coluna de entrada, cada uma

contendo um para todos os valores pares (ou seja, o módulo de valor dois é zero) ou um zero para todos os valores não pares (isto é, valores ímpares).

Nota: Para valores de número não inteiro, a função macro INT é aplicada primeiro. Por exemplo, ISEVEN(2.5) = 1, já que 2 é par.

Exemplos

TEMP = ISEVEN(-3) Cria uma nova coluna denominada TEMP contendo o valor zero.
TEMP = ISEVEN(V1) Cria uma nova coluna denominada TEMP, em que cada valor é o resultado do teste do conteúdo da coluna V1 para detectar valores pares.
TEMP = ISEVEN(V1:V3) Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os resultados do teste do conteúdo da coluna V1 para detectar valores pares, os valores da coluna VX são os resultados do teste do conteúdo da coluna V2 para detectar valores pares, e os valores da coluna VY são os resultados do teste do conteúdo da coluna V3 para detectar valores pares.
TEMP = ISEVEN(V1[10:20]) Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm os resultados dos testes dos valores nas linhas 10-20 da coluna V1 para detectar valores pares. Outras células em TEMP estão vazias.
TEMP = ISEVEN(V1[1:5]:V2) Cria duas novas colunas denominadas TEMP e VX, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna TEMP são os resultados do teste das linhas correspondentes da coluna V1 para detectar valores pares, e os valores na coluna VX são os resultados do teste das linhas correspondentes da coluna V2 para detectar valores pares.

Funções relacionadas

Função	Descrição
ISODD	Testa se os valores de entrada são ímpares (ou seja, não divisíveis por dois)

Macro ISODD

O ISODD está disponível apenas no IBM Campaign.

Sintaxe

ISODD(data)

Parâmetros

data

Os valores numéricos para testar se eles são ímpares. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção

"Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

ISODD testa se cada valor no conjunto de dados especificado é ímpar. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo um para todos os valores ímpares (ou seja, o módulo de valor dois é um) ou um zero para todos os valores não ímpares (isto é, valores pares).

Nota: Para valores de número não inteiro, a função macro INT é aplicada primeiro. Por exemplo, ISODD(2.5) = 0, já que 2 não é ímpar.

Exemplos

TEMP = ISODD(-3) Cria uma nova coluna denominada TEMP contendo o valor um.
TEMP = ISODD(V1) Cria uma nova coluna denominada TEMP, em que cada valor é o resultado do teste do conteúdo da coluna V1 para detectar valores ímpares.
TEMP = ISODD(V1:V3) Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os resultados do teste do conteúdo da coluna V1 para detectar valores ímpares, os valores da coluna VX são os resultados do teste do conteúdo da coluna V2 para detectar valores ímpares, e os valores da coluna VY são os resultados do teste do conteúdo da coluna V3 para detectar valores ímpares.
TEMP = ISODD(V1[10:20]) Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm os resultados dos testes dos valores nas linhas 10-20 da coluna V1 para detectar valores ímpares. Outras células em TEMP estão vazias.
TEMP = ISODD(V1[1:5]:V2) Cria duas novas colunas denominadas TEMP e VX, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna TEMP são os resultados do teste das linhas correspondentes da coluna V1 para detectar valores ímpares, e os valores na coluna VX são os resultados do teste das linhas correspondentes da coluna V2 para detectar valores ímpares.

Funções relacionadas

Função	Descrição
ISEVEN	Testa se os valores de entrada são pares (ou seja, divisíveis por dois)

Macro LE

O LE está disponível no IBM Campaign e no IBM Interact.

Sintaxe

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

Parâmetros

`data1`

O intervalo de células numéricas para comparar. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`data2`

O número ou os números com relação aos quais comparar todos os valores na coluna especificada. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`LE` compara os dois intervalos de dados especificados, retornando um se os valores no primeiro conjunto de dados forem menores ou iguais aos valores no segundo conjunto de dados ou, caso contrário, um zero. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em `data1` em comparação com a coluna correspondente de `data2` (ou seja, a primeira coluna de `data1` é comparada com a primeira coluna de `data`, a segunda coluna com a segunda coluna, e assim por diante).

Se `data2` for uma constante, cada valor em `data` será comparado com esse valor. Se `data2` for uma coluna, os cálculos serão executados em uma base de linha por linha. Os valores em `data1` são comparados com o valor da primeira linha de `data2`, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: O operador `LE` pode ser abreviado com um sinal de menor que seguido por um sinal de igual (`<=`).

Exemplos

`TEMP = 4 LE 4` ou `TEMP = 4 <= 4` Cria uma nova coluna denominada `TEMP` contendo o valor um (já que quatro é igual a ele mesmo).
`TEMP = V1 <= 8` Cria uma nova coluna denominada `TEMP`, em que cada valor será um se o valor da linha correspondente da coluna `V1` for menor ou igual ao número oito, caso contrário, zero.
`TEMP = V1:V3 <= 2` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são o conteúdo da coluna `V1` em comparação com o valor dois, os valores da coluna `VX` são o conteúdo da coluna `V2` em comparação com o valor dois, e os valores da coluna `VY` são o conteúdo da coluna `V3` em comparação com o valor dois.
`TEMP = V1 <= V1` Cria uma nova coluna denominada `TEMP` contendo todos os números um (já que cada número é igual a ele mesmo).

`TEMP = V1 <= V2` Cria uma nova coluna denominada `TEMP`, em que cada valor é o valor da linha da coluna `V1` em comparação com o valor da linha correspondente da coluna `V2`.
`TEMP = V1[10:20] <= V2` ou `TEMP = V1[10:20] <= V2[1:11]` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os resultados da comparação dos valores nas linhas 10-20 da coluna `V1` com os valores nas linhas 1-11 da coluna `V2`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`EQ`	Retornará TRUE se um intervalo de dados for igual ao outro
`GE`	Retornará TRUE se um intervalo de dados for maior ou igual ao outro
`GT`	Retornará TRUE se um intervalo de dados for maior que o outro
`LT`	Retornará TRUE se um intervalo de dados for menor que o outro
`NE`	Retornará TRUE se um intervalo de dados não for igual ao outro

Macro LIKE

O `LIKE` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
data1 [NOT] LIKE data2
```

Parâmetros

`data1`

O intervalo de células para comparar. Esta pode ser uma sequência de texto ou uma expressão avaliada como uma sequência de texto. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`data2`

O padrão de texto com relação ao qual comparar todos os valores na coluna especificada. Esta pode ser uma sequência de texto ou uma expressão avaliada como uma sequência de texto. O número de colunas em `data2` deve ser igual ao número de colunas em `data1`, a menos que `data2` seja uma constante. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Um sublinhado (_) em `data2` representa um caractere curinga que corresponderá a qualquer caractere único em `data1`. Um sinal de porcentagem (%) corresponderá a zero ou mais caracteres em `data1`.

Descrição

LIKE compara os dois intervalos de dados especificados, retornando um se as sequências corresponderem ou zero se não corresponderem. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em data1 em comparação com a coluna correspondente de data2 (ou seja, a primeira coluna de data1 é comparada com a primeira coluna de data2, a segunda coluna com a segunda coluna, e assim por diante).

Se data2 for uma constante da sequência, cada sequência em data1 será comparada com essa sequência. Se data2 for uma coluna, os cálculos serão executados em uma base de linha por linha. A primeira sequência de linha em data1 é comparada com a primeira sequência da linha de data2, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até a última sequência da coluna mais curta.

Ao comparar sequências, maiúsculas e minúsculas não importam (ou seja, "Yes", "YES", "yes" e "yeS" são todos considerados iguais).

Nota: A macro LIKE possui uma versão negativa, NOT LIKE. O formato para isso é idêntico ao LIKE. NOT LIKE retornará um se a sequência em data1 não corresponder ao modelo definido por data2.

Exemplos

`TEMP = "gold" LIKE "gold"` Cria uma nova coluna denominada TEMP contendo o valor um (desde que as duas sequências correspondam).
`TEMP = "No" LIKE "NO"` Cria uma nova coluna denominada TEMP contendo o valor um (as comparações de sequências não fazem distinção entre maiúsculas e minúsculas).
`TEMP = V1 LIKE "gold%"` Cria uma nova coluna denominada TEMP, em que cada valor será um se o valor da linha correspondente da coluna V1 for igual à sequência "gold" seguida por qualquer número de caracteres. Caso contrário, cada valor será zero.
`TEMP = V1 LIKE "g_ld"` Cria uma nova coluna denominada TEMP, em que cada valor será um se o valor de linha correspondente da coluna V1 for igual à sequência "g" seguida por qualquer caractere e seguida por "ld". Caso contrário, cada valor será zero.
`TEMP = V1 LIKE V1` Cria uma nova coluna denominada TEMP contendo todos os números um (já que cada número é igual a ele mesmo).
`TEMP = V1 LIKE V2` Cria uma nova coluna denominada TEMP, em que cada valor é o valor da linha da coluna V1 em comparação com o valor da linha correspondente da coluna V2.
`TEMP = V1:V3 LIKE V4:V6` Cria três novas colunas denominadas TEMP, VX e VY. A coluna TEMP contém as sequências em V1 em comparação com as sequências de linha correspondente da coluna V4. A coluna VX compara as colunas V2 e V5. A coluna VY compara as colunas V3 e V6.

`TEMP = V1[10:20] LIKE V2` ou `TEMP = V1[10:20] LIKE V2[1:11]` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os resultados da comparação das sequências nas linhas 10-20 da coluna `V1` com as linhas 1-11 da coluna `V2`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`EQ`	Retornará TRUE se um intervalo de dados for igual ao outro
`GE`	Retornará TRUE se um intervalo de dados for maior ou igual ao outro
`GT`	Retornará TRUE se um intervalo de dados for maior que o outro
`LE`	Retornará TRUE se um intervalo de dados for menor ou igual ao outro
`LT`	Retornará TRUE se um intervalo de dados for menor que o outro
`NE`	Retornará TRUE se um intervalo de dados não for igual ao outro

Macro LN ou LOG

O `LN` ou `LOG` está disponível apenas no IBM Campaign.

Sintaxe

`LN(data)` ou `LOG(data)`

Parâmetros

`data`

Os valores numéricos dos quais calcular o logaritmo natural. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`LN` ou `LOG` calcula o log natural de cada valor no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo o logaritmo natural dos números na coluna de entrada correspondente. Os logaritmos naturais baseiam-se na constante e = 2,7182818. `LN` é o inverso da função macro `EXP`.

Nota: Todos os valores no intervalo de dados especificado devem ser maiores que zero. Caso contrário, uma célula em branco será retornada para cada entrada inválida.

Exemplos

`TEMP = LN(3)` ou `TEMP = LOG(3)` Cria uma nova coluna denominada `TEMP` contendo o valor `1.099`.
`TEMP = LN(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é o log natural do conteúdo da coluna `V1`.
`TEMP = LN(V1:V3)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os logs naturais do conteúdo da coluna `V1`, os valores na coluna `VX` são os logs naturais do conteúdo da coluna `V2`, e os valores na coluna `VY` são os logs naturais do conteúdo da coluna `V3`.
`TEMP = LN(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os logs naturais dos valores nas linhas 10-20 da coluna `V1`. Outras células em `TEMP` estão vazias.
`TEMP = LN(V1[1:5]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são os logs naturais das linhas correspondentes da coluna `V1`, e os valores na coluna `VX` são os logs naturais das linhas correspondentes da coluna `V2`.

Funções relacionadas

Função	Descrição
`EXP`	Calcula um ou mais números naturais elevados ao conteúdo de cada célula no intervalo de dados especificado
`LOG2`	Calcula o log de base2 do conteúdo do intervalo de dados especificado
`LOG10`	Calcula o log de base10 do conteúdo do intervalo de dados especificado
`POW`	Calcula um valor base elevado a uma ou mais potências exponenciais especificadas

Macro LOG2

O `LOG2` está disponível apenas no IBM Campaign.

Sintaxe

```
LOG2(data)
```

Parâmetros

`data`

Os valores numéricos dos quais calcular o logaritmo base2. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

LOG2 calcula o logaritmo de base 2 dos valores no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo o logaritmo base2 dos números na coluna de entrada correspondente.

Nota: Todos os valores no intervalo de dados especificado devem ser maiores que zero. Caso contrário, uma célula em branco será retornada para cada entrada inválida.

Exemplos

`TEMP = LOG2(8)` Cria uma nova coluna denominada TEMP contendo o valor três.
`TEMP = LOG2(V1)` Cria uma nova coluna denominada TEMP, em que cada valor é o log base2 do conteúdo da coluna V1.
`TEMP = LOG2(V1:V3)` Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os logs base2 do conteúdo da coluna V1, os valores da coluna VX são os logs base2 do conteúdo de sua coluna V2, e os valores da coluna VY são os logs base2 do conteúdo da coluna V3.
`TEMP = LOG2(V1[10:20])` Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm os logs de base 2 dos valores nas linhas 10-20 da coluna V1. Outras células em TEMP estão vazias.
`TEMP = LOG2(V1[1:5]:V2)` Cria duas novas colunas denominadas TEMP e VX, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna TEMP são os logs de base 2 das linhas correspondentes da coluna V1, e os valores na coluna VX são os logs de base 2 das linhas correspondentes da coluna V2.

Funções relacionadas

Função	Descrição
LN ou LOG	Calcula o log natural do conteúdo do intervalo de dados especificado
LOG10	Calcula o log de base10 do conteúdo do intervalo de dados especificado
POW	Potência exponencial

Macro LOG10

O LOG10 está disponível apenas no IBM Campaign.

Sintaxe

```
LOG10(data)
```

Parâmetros

data

Os valores numéricos dos quais calcular o logaritmo base10. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`LOG10` calcula o logaritmo de base 10 dos valores no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo o logaritmo base10 dos números na coluna de entrada correspondente.

Nota: Todos os valores no intervalo de dados especificado devem ser maiores que zero. Caso contrário, uma célula em branco será retornada para cada entrada inválida.

Exemplos

`TEMP = LOG10(100)` Cria uma nova coluna denominada `TEMP` contendo o valor dois.
`TEMP = LOG10(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é o log base10 do conteúdo da coluna `V1`.
`TEMP = LOG10(V1:V3)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os logs base10 do conteúdo da coluna `V1`, os valores da coluna `VX` são os logs base10 do conteúdo de sua coluna `V2`, e os valores da coluna `VY` são os logs base10 do conteúdo da coluna `V3`.
`TEMP = LOG10(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os logs de base 10 dos valores nas linhas 10-20 da coluna `V1`. Outras células em `TEMP` estão vazias.
`TEMP = LOG10(V1[1:5]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são os logs de base 10 das linhas correspondentes da coluna `V1`, e os valores na coluna `VX` são os logs de base 10 das linhas correspondentes da coluna `V2`.

Funções relacionadas

Função	Descrição
`LN` ou `LOG`	Calcula o log natural do conteúdo do intervalo de dados especificado
`LOG2`	Calcula o log de base2 do conteúdo do intervalo de dados especificado
`POW`	Potência exponencial

Macro LOWER

O `LOWER` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
LOWER(data)
```

Parâmetros

`data`

O valor de sequência a ser convertido em minúsculas.

Descrição

`LOWER` converte cada valor de sequência no intervalo de dados especificado em minúsculas. Ele retorna uma nova coluna com cada célula contendo a sequência em minúsculas da célula de entrada correspondente.

Exemplos

`Temp = LOWER "GOLD"` Cria uma nova coluna denominada Temp contendo "gold".
`TEMP = LOWER( "JAN 15, 1997")` Cria uma nova coluna denominada `TEMP`, que contém a sequência de texto ASCII " `jan 15, 1997` ".
`TEMP = LOWER( "Pressure")` Cria uma nova coluna denominada `TEMP`, que contém a sequência de texto ASCII `"pressure"`.
`TEMP = LOWER(V1)` Cria uma nova coluna denominada `TEMP` contendo caracteres minúsculos de cada sequência na coluna `V1`.

Macro LT

O `LT` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
data1 LT data2 data1 < data2
```

Parâmetros

`data1`

O intervalo de células numéricas para comparar. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`data2`

O número ou os números com relação aos quais comparar todos os valores na coluna especificada. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a

definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

LT compara os dois intervalos de dados especificados, retornando um se os valores no primeiro conjunto de dados forem menores que os valores no segundo conjunto de dados ou, caso contrário, um zero. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em data1 em comparação com a coluna correspondente de data2 (ou seja, a primeira coluna de data1 é comparada com a primeira coluna de data, a segunda coluna com a segunda coluna, e assim por diante).

Se data2 for uma constante, cada valor em data será comparado com esse valor. Se data2 for uma coluna, os cálculos serão executados em uma base de linha por linha. Os valores em data1 são comparados com o valor da primeira linha de data2, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: O operador LT pode ser abreviado com um sinal de menor que (<).

Exemplos

TEMP = 3 LT 4 ou TEMP = 3 < 4 Cria uma nova coluna denominada TEMP contendo o valor um (já que três é menor que quatro).
TEMP = V1 < 8 Cria uma nova coluna denominada TEMP, em que cada valor será um se o valor da linha correspondente da coluna V1 for menor que o número oito, caso contrário, zero.
TEMP = V1:V3 < 2 Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são o conteúdo da coluna V1 em comparação com o valor dois, os valores da coluna VX são o conteúdo da coluna V2 em comparação com o valor dois, e os valores da coluna VY são o conteúdo da coluna V3 em comparação com o valor dois.
TEMP = V1 < V1 Cria uma nova coluna denominada TEMP contendo todos os zeros (já que nenhum número é menor que ele mesmo).
TEMP = V1 < V2 Cria uma nova coluna denominada TEMP, em que cada valor é o valor da linha da coluna V1 em comparação com o valor da linha correspondente da coluna V2.
TEMP = V1[10:20] < V2 ou TEMP = V1[10:20] < V2[1:11] Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm os resultados da comparação dos valores nas linhas 10-20 da coluna V1 com as linhas 1-11 da coluna V2. Outras células em TEMP estão vazias.

Funções relacionadas

Função	Descrição
EQ	Retornará TRUE se um intervalo de dados for igual ao outro
GE	Retornará TRUE se um intervalo de dados for maior ou igual ao outro
GT	Retornará TRUE se um intervalo de dados for maior que o outro
LE	Retornará TRUE se um intervalo de dados for menor ou igual ao outro

Macro LTRIM

O LTRIM está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
LTRIM(data)
```

Parâmetros

data

A sequência a partir da qual o espaço à esquerda será removido.

Descrição

LTRIM remove caracteres de espaço à esquerda de cada valor de sequência no intervalo de dados especificado, retornando a sequência convertida. Ele retorna uma nova coluna para cada coluna de entrada.

Exemplos

```
Temp = LTRIM " gold"
```

Cria uma nova sequência denominada Temp contendo "gold".

Macro MAX

O MAX está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
MAX(data [, keyword])
```

Parâmetros

data

Os valores numéricos dos quais calcular o máximo. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

keyword

Essa palavra-chave opcional determina como o cálculo será executado sobre o intervalo de dados de entrada. Selecione uma das opções a seguir:

ALL - Executa o cálculo em todas as células em data (padrão)

COL - Executa o cálculo separadamente para cada coluna de data

ROW - Executa o cálculo separadamente para cada linha de data

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte "Especificações de Formato" na página 9.

Nota: Muitas funções de macro utilizam os parâmetros de palavra-chave {ALL | COL | ROW}. Estas palavras-chave não se aplicam ao IBM Campaign porque os dados de entrada são sempre uma coluna ou um campo único. A macro sempre se comportará como se a palavra-chave COL fosse especificada. Portanto, não é necessário especificar estas palavras-chave ao usar o IBM Campaign.

Descrição

MAX calcula o máximo dos valores no intervalo de dados especificado. Ele retorna uma nova coluna única contendo o valor máximo.

Exemplos

TEMP = MAX(3) ou TEMP = MAX(3, ALL) Cria uma nova coluna denominada TEMP contendo o valor três.
TEMP = MAX(V1) Cria uma nova coluna denominada TEMP contendo um valor único que é o valor máximo do conteúdo da coluna V1.
TEMP = MAX(V1:V3) Cria uma nova coluna denominada TEMP contendo um valor único que é o máximo das colunas V1, V2 e V3.
TEMP = MAX(V1[10:20]) Cria uma nova coluna denominada TEMP contendo um valor único que é o máximo das células nas linhas 10-20 da coluna V1.
TEMP = MAX(V1[1:5]:V4) Cria uma nova coluna denominada TEMP contendo um valor único que é o máximo das células nas linhas 1-5 das colunas V1 a V4.
TEMP = MAX(V1:V3, COL) Cria três novas colunas denominadas TEMP, VX e VY. O único valor na coluna TEMP é o máximo do conteúdo da coluna V1, o valor único na coluna VX é o máximo do conteúdo da coluna V2, e o único valor na coluna VY é o máximo do conteúdo da coluna V3.
TEMP = MAX(V1[1:5]:V3, COL) Cria três novas colunas denominadas TEMP, VX e VY, cada uma contendo um valor único. O valor na coluna TEMP é o máximo das células nas linhas 1-5 da coluna V1, o valor na coluna VX é o máximo das células nas linhas 1-5 da coluna V2,, e o valor na coluna VY é o máximo das células nas linhas 1-5 da coluna V3.

`TEMP = MAX(V1:V3, ROW)` Cria uma nova coluna denominada `TEMP` em que cada entrada na célula é o máximo da linha correspondente nas colunas `V1`, `V2` e `V3`.
`TEMP = MAX(V1[10:20]:V3, ROW)` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm o máximo dos valores nas linhas 10-20 entre as colunas `V1` e `V3`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`MIN`	Calcula o mínimo de um intervalo de células

Macro MEAN

O `MEAN` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
MEAN(data [, keyword])
```

Parâmetros

`data`

Os valores numéricos dos quais calcular a média aritmética. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`keyword`

Essa palavra-chave opcional determina como o cálculo será executado sobre o intervalo de dados de entrada. Selecione uma das opções a seguir:

`ALL` - Executa o cálculo em todas as células em `data` (padrão)

`COL` - Executa o cálculo separadamente para cada coluna de `data`

`ROW` - Executa o cálculo separadamente para cada linha de `data`

Consulte “Macro DATE” na página 51 para obter mais detalhes sobre como utilizar palavras-chaves.

Nota: Muitas funções de macro utilizam os parâmetros de palavra-chave {`ALL` | `COL` | `ROW`}. Estas palavras-chave não se aplicam ao IBM Campaign porque os dados de entrada são sempre uma coluna ou um campo único. A macro sempre se comportará como se a palavra-chave `COL` fosse especificada. Portanto, não é necessário especificar estas palavras-chave ao usar o IBM Campaign.

Descrição

MEAN calcula a média aritmética ou a média das células no intervalo de dados especificado. A média aritmética é calculada ao somar o conteúdo de todas as células dividido pelo número de células. O número de colunas retornadas por MEAN depende da keyword.

- Se keyword for ALL, MEAN retornará uma nova coluna, contendo um valor único (a média de todas as células em data).
- Se keyword for COL, MEAN retornará uma nova coluna para cada coluna de entrada. Cada nova coluna contém um valor (a média de todas as células na coluna de entrada correspondente).
- Se keyword for ROW, MEAN retornará uma nova coluna contendo a média de cada linha de data.

Nota: Células em branco são ignoradas na média.

Nota: MEAN é o mesmo que a função macro AVG.

Exemplos

`TEMP = MEAN(V1)` Cria uma nova coluna denominada TEMP contendo um valor único que é a média aritmética do conteúdo da coluna V1.
`TEMP = MEAN(V1:V3)` Cria uma nova coluna denominada TEMP contendo um valor único que é a média aritmética do conteúdo das colunas V1, V2 e V3.
`TEMP = MEAN(V1[10:20])` Cria uma nova coluna denominada TEMP contendo um valor único que é a média aritmética das células nas linhas 10-20 da coluna V1.
`TEMP = MEAN(V1[1:5]:V4)` Cria uma nova coluna denominada TEMP contendo um valor único que é a média aritmética das células nas linhas 1-5 das colunas V1 a V4.
`TEMP = MEAN(V1:V3, COL)` Cria três novas colunas denominadas TEMP, VX e VY. O valor único na coluna TEMP é a média aritmética do conteúdo da coluna V1, o valor único na coluna VX é a média aritmética do conteúdo da coluna V2, e o valor único na coluna VY é a média aritmética do conteúdo da coluna V3.
`TEMP = MEAN(V1[10:20]:V3, COL)` Cria três novas colunas denominadas TEMP, VX e VY, cada uma contendo um valor único. O valor na coluna TEMP é a média aritmética das células nas linhas 10-20 da coluna V1, o valor na coluna VX é a média aritmética das células nas linhas 10-20 da coluna V2, e o valor na coluna VY é a média aritmética das células nas linhas 10-20 da coluna V3.
`TEMP = MEAN(V1:V3, ROW)` Cria uma nova coluna denominada TEMP, em que cada entrada na célula é a média aritmética da linha correspondente nas colunas V1, V2e V3.

`TEMP = MEAN(V1[1:5]:V3,ROW)` Cria uma nova coluna denominada `TEMP`, em que as células nas linhas 1-5 contêm a média aritmética da linha correspondente nas colunas `V1` a `V3`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`SUM` ou `TOTAL`	Calcula a soma de um intervalo de células

Macro MIN

A macro MIN está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
MIN(data [, keyword])
```

Parâmetros

`data`

Os valores numéricos dos quais calcular o mínimo. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`keyword`

Essa palavra-chave opcional determina como o cálculo será executado sobre o intervalo de dados de entrada. Selecione uma das opções a seguir:

`ALL` - Executa o cálculo em todas as células em `data` (padrão)

`COL` - Executa o cálculo separadamente para cada coluna de `data`

`ROW` - Executa o cálculo separadamente para cada linha de `data`

Consulte “Macro DATE” na página 51 para obter mais detalhes sobre como utilizar palavras-chaves.

Nota: Muitas funções de macro utilizam os parâmetros de palavra-chave {`ALL` | `COL` | `ROW`}. Estas palavras-chave não se aplicam ao **IBM Campaign** porque os dados de entrada são sempre uma coluna ou um campo único. A macro sempre se comportará como se a palavra-chave `COL` fosse especificada. Portanto, não é necessário especificar estas palavras-chave quando usar **IBM Campaign** .

Descrição

`MIN` calcula o mínimo de todas as células no intervalo de dados especificado. Ele retorna uma coluna única contendo o valor mínimo.

Exemplos

`TEMP = MIN(V1)` Cria uma nova coluna denominada `TEMP` contendo um valor único que é o valor mínimo da coluna `V1`.
`TEMP = MIN(V1:V3)` Cria uma nova coluna denominada `TEMP` contendo um valor único que é o mínimo de colunas `V1`, `V2` e `V3`.
`TEMP = MIN(V1[10:20])` Cria uma nova coluna denominada `TEMP` contendo um valor único que é o mínimo das células nas linhas 10-20 da coluna `V1`.
`TEMP = MIN(V1[1:5]:V4)` Cria uma nova coluna denominada `TEMP` contendo um valor único que é o mínimo das células nas linhas 1-5 das colunas `V1` a `V4`.
`TEMP = MIN(V1:V3, COL)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. O valor único na coluna `TEMP` é o mínimo da coluna `V1`, o valor único na coluna `VX` é o mínimo da coluna `V2`, e o valor único da coluna `VY` é o mínimo da coluna `V3`.
`TEMP = MIN(V1[1:5]:V3, COL)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`, cada uma contendo um valor único. O valor na coluna `TEMP` é o mínimo das células nas linhas 1-5 da coluna `V1`, o valor na coluna `VX` é o mínimo das células nas linhas 1-5 da coluna `V2`, e o valor na coluna `VY` é o mínimo das células nas linhas 1-5 da coluna `V3`.
`TEMP = MIN(V1:V3, ROW)` Cria uma nova coluna denominada `TEMP`, em que cada entrada na célula é o mínimo da linha correspondente nas colunas `V1`, `V2` e `V3`.
`TEMP = MIN(V1[10:20]:V3, ROW)` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm o mínimo dos valores nas linhas 1-5 entre as colunas `V1` e `V3`. Outras células em `TEMP` estão vazias.

Funções Relacionadas

Função	Descrição
`MAX`	Calcula o máximo de um intervalo de células
`MAX_TO_INDEX`	Retorna o índice da coluna do valor máximo para cada linha da coluna especificada

Macro MINUS

O `MINUS` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
data MINUS subtrahend data - subtrahend
```

Parâmetros

`data`

O intervalo de células contendo números dos quais subtrair. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`subtrahend`

Um ou mais números para subtrair de todos os valores na coluna especificada. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em `subtrahend` deve ser igual ao número de colunas em `data`, a menos que `subtrahend` seja uma constante. Para obter a definição de formato de `subtrahend` (mesmo que `data`), consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`MINUS` subtrai o `subtrahend` do intervalo de dados especificado `data`. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em `data` menos a coluna correspondente de `subtrahend` (ou seja, a primeira coluna de `data` subtrai a primeira coluna de `subtrahend`, a segunda coluna com a segunda coluna, e assim por diante).

Se `subtrahend` for uma constante, cada valor em `data` será subtraído deste valor. Se `subtrahend` contiver uma ou mais colunas, os cálculos serão executados em uma base de linha por linha entre uma coluna `data` e uma coluna de `subtrahend`. A primeira linha de `data` subtrai o valor da primeira linha de `subtrahend`, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: O operador `MINUS` pode ser abreviado com um sinal de menos ou hífen (`-`).

Exemplos

`TEMP = 7 MINUS 4` ou `TEMP = 7 - 4` Cria uma nova coluna denominada `TEMP` contendo o valor três.
`TEMP = V1 - 8` Cria uma nova coluna denominada `TEMP`, em que cada valor é o conteúdo da coluna `V1` menos oito.
`TEMP = V1:V3 - 2` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são o conteúdo da coluna `V1` menos dois, os valores da coluna `VX` são o conteúdo da coluna `V2` menos dois, e os valores da coluna `VY` são o conteúdo da coluna `V3` menos dois.
`TEMP = V1 - V1` Cria uma nova coluna denominada `TEMP` contendo todos zeros (já que qualquer número de menos ele mesmo é zero).

`TEMP = V1 - V2` Cria uma nova coluna denominada `TEMP`, em que cada valor é o valor da linha da coluna `V1` menos o valor da linha correspondente da coluna `V2`.
`TEMP = V1:V3 -V4:V6` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. A coluna `TEMP` contém os valores em `V1` menos os valores da linha correspondente da coluna `V4`. A coluna `VX` subtrai a coluna `V5` de `V2`. A coluna `VY` subtrai a coluna `V6` de `V3`.
`TEMP = V1[10:20] - V2` ou `TEMP = V1[10:20] - V2[1:11]` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os valores nas linhas 10-20 da coluna `V1` menos os valores nas linhas 1-11 da coluna `V2`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`PLUS`	Soma o conteúdo de dois intervalos de dados
`SUM` ou `TOTAL`	Calcula a soma de um intervalo de células

Macro MOD

O `MOD` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
data MOD divisor data % divisor
```

Parâmetros

`data`

Os valores de números inteiros a partir dos quais calcular o módulo. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`divisor`

O número inteiro de base diferente de zero com relação ao qual calcular o módulo. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em `divisor` deve ser igual ao número de colunas em `data`, a menos que `divisor` seja uma constante. Para obter a definição de formato de `divisor` (mesmo que `data`), consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`MOD` calcula o resto da divisão do intervalo de dados especificado por um valor especificado. Isto é calculado dividindo-se o `divisor` por cada valor e retornando o

resto. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo os números no divisor do módulo data. O resto terá o mesmo sinal (positivo ou negativo) como data.

Se divisor for uma constante, cada valor na coluna especificada será calculado no módulo desse valor. Se divisor for uma coluna, os cálculos serão executados em uma base de linha por linha. Os valores em data são calculados no módulo do primeiro valor da linha de divisor, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: Se divisor for zero, um erro de divisão por zero será retornado.

Nota: O operador MOD pode ser abreviado com o sinal de porcentagem (%). Por exemplo, TEMP = 5 % 3 é equivalente a TEMP = 5 MOD 3.

Exemplos

TEMP = 10 MOD 8 ou TEMP = 10 % 8 Cria uma nova coluna denominada TEMP contendo o valor 2.
TEMP = -10 % 8 Cria uma nova coluna denominada TEMP contendo o valor -2.
TEMP = V1 % 8 Cria uma nova coluna denominada TEMP, em que cada valor é o conteúdo da coluna V1, módulo oito.
TEMP = V1:V3 % 2 Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os valores do módulo dois do conteúdo da coluna V1, os valores da coluna VX são os valores do módulo dois do conteúdo da coluna V2, e os valores da coluna VY são os valores do módulo dois do conteúdo da coluna V3.
TEMP = V1 % V1 Cria uma nova coluna denominada TEMP, contendo um zero para cada entrada na coluna V1. Isso ocorre porque cada módulo de número dele mesmo é zero.
TEMP = V1 % V2 Cria uma nova coluna denominada TEMP, em que cada valor é o valor da linha da coluna V1 no módulo do valor da linha correspondente da coluna V2. Observe que, se V1=V2, então todos os zeros serão retornados, como no exemplo anterior.
TEMP = V1:V3 % V4:V6 Cria três novas colunas denominadas TEMP, VX e VY. A coluna TEMP contém os valores em V1 no módulo dos valores da linha correspondente da coluna V4. A coluna VX contém os resultados da coluna V2 no módulo V5. A coluna VY contém os resultados da coluna V3 no módulo V6.
TEMP = V1[10:20] % V2 ou TEMP = V1[10:20] % V2[1:11] Cria uma nova coluna denominada TEMP, em que as 11 primeiras células são os valores nas linhas 10-20 da coluna V1 no módulo dos valores nas linhas 1-11 da coluna V2. Outras células em TEMP estão vazias.

Funções relacionadas

Função	Descrição
DIV	Divide um intervalo de dados especificado por outro
MOD	Calcula o módulo do conteúdo do intervalo de dados especificado

Macro MONTHOF

O MONTHOF está disponível apenas no IBM Campaign.

Sintaxe

```
MONTHOF(date_string [, input_format])
```

Parâmetros

date_string

Um texto representando uma data válida.

input_format

Uma das palavras-chave na tabela abaixo, especificando o formato de data de date_string.

Descrição

MONTHOF retorna o mês como um número para a data especificada pelo date_string. Se input_format não for fornecido, a palavra-chave padrão DELIM_M_D_Y será utilizada.

Exemplos

MONTHOF("012171",MMDDAA) retorna o número 1.

Nota: Consulte "Macro DATE" na página 51 para obter informações adicionais sobre formatos de data válidos.

Funções relacionadas

Função	Descrição
DAYOF	Retorna o dia da semana como um número.
WEEKDAYOF	Retorna o dia da semana como um número.
YEAROF	Retorna o ano como um número.

Macro MULT

O MULT está disponível no IBM Campaign e no IBM Interact.

Sintaxe

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

Parâmetros

`data`

Os valores numéricos para multiplicar. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`multiplier`

O número pelo qual multiplicar todos os valores na coluna especificada. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em `multiplier` deve ser igual ao número de colunas em `data`, a menos que `multiplier` seja uma constante. Para obter a definição de formato de `multiplier` (mesmo que `data`), consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`MULT` multiplica os valores nos dois intervalos de dados especificados. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo os números em `data` multiplicados por `multiplier`. Se `multiplier` for uma constante, cada valor em `data` será multiplicado por este valor. Se `multiplier` for uma coluna, os cálculos serão executados em uma base de linha por linha. Os valores em `data` são multiplicados pelo valor da primeira linha de `multiplier`, a segunda linha com a segunda linha e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: Usar uma coluna contendo o mesmo número *x* em cada linha como `multiplier` é o mesmo que utilizar a constante *x* como `multiplier`.

Nota: O operador `MULT` pode ser abreviado com um asterisco (`*`).

Exemplos

`TEMP = 8 MULT 4` ou `TEMP = 8 * 4` Cria uma nova coluna denominada `TEMP` contendo o valor `32`.
`TEMP = V1 * 8` Cria uma nova coluna denominada `TEMP`, em que cada valor é o conteúdo da coluna `V1` multiplicado por oito.
`TEMP = V1:V3 * 2` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são duas vezes o conteúdo da coluna `V1`, os valores da coluna `VX` são duas vezes o conteúdo da coluna `V2`, e os valores da coluna `VY` são duas vezes o conteúdo da coluna `V3`.

`TEMP = V1 * V1` Cria uma nova coluna denominada `TEMP` que contém o quadrado de cada valor na coluna `V1`.
`TEMP = V1 * V2` Cria uma nova coluna denominada `TEMP`, em que cada valor é o valor da linha da coluna `V1` multiplicado pelo valor da linha correspondente da coluna `V2`.
`TEMP = V1:V3 * V4:V6` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. A coluna `TEMP` contém os valores em `V1` vezes os valores da linha correspondente da coluna `V4`. A coluna `VX` multiplica a coluna `V2` pela `V5`. A coluna `VY` multiplica a coluna `V3` pela `V6`.
`TEMP = V1[10:20] * V2` ou `TEMP = V1[10:20] * V2[1:11]` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os valores nas linhas 10-20 da coluna `V1` vezes os valores nas linhas 1-11 da coluna `V2`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`DIV`	Divide um intervalo de dados especificado por outro
`EXP`	Calcula um ou mais números naturais elevados ao conteúdo de cada célula no intervalo de dados especificado
`POW`	Calcula um valor base elevado a uma ou mais potências exponenciais especificadas

Macro NE

O `NE` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
data1 NE data2 data1 != data2 data1 <> data2
```

Parâmetros

`data1`

O intervalo de células para comparar. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`data2`

O número ou os números com relação aos quais comparar todos os valores na coluna especificada. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em `data2` deve ser igual ao número de colunas em `data1`, a menos que `data2` seja uma constante. Para obter a definição de formato de `data`, consulte a

seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

NE compara os dois intervalos de dados especificados, retornando um se os valores não forem iguais ou um zero se forem iguais. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em data1 em comparação com a coluna correspondente de data2 (ou seja, a primeira coluna de data1 é comparada com a primeira coluna de data, a segunda coluna com a segunda coluna, e assim por diante).

Se data2 for uma constante, cada valor em data1 será comparado com este valor. Se data2 for uma coluna, os cálculos serão executados em uma base de linha por linha. Os valores na primeira linha de data1 são comparados com o valor da primeira linha de data2, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: Usar uma coluna contendo o mesmo número *x* em cada linha como data2 é o mesmo que utilizar a constante *x* como data2.

Nota: O operador NE pode ser abreviado com um ponto de exclamação seguido por um sinal de igual (!=) ou por um sinal de menor que seguido por um sinal de maior que (<>).

Exemplos

TEMP = 3 NE 4 ou TEMP = 3 != 4 TEMP = 3 <> 4 Cria uma nova coluna denominada TEMP contendo o valor um (já que três não é igual a quatro).
TEMP = V1 != 8 Cria uma nova coluna denominada TEMP, em que cada valor é um se o valor da linha correspondente da coluna V1 não for igual ao número oito, caso contrário, zero.
TEMP = V1:V3 != 2 Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são o conteúdo da coluna V1 em comparação com o valor dois, os valores da coluna VX são o conteúdo da coluna V2 em comparação com o valor dois, e os valores da coluna VY são o conteúdo da coluna V3 em comparação com o valor dois.
TEMP = V1 != V1 Cria uma nova coluna denominada TEMP contendo todos os zeros (já que cada número é igual a ele mesmo).
TEMP = V1 != V2 Cria uma nova coluna denominada TEMP, em que cada valor é o valor da linha da coluna V1 em comparação com o valor da linha correspondente da coluna V2.
TEMP = V1:V3 != V4:V6 Cria três novas colunas denominadas TEMP, VX e VY. A coluna TEMP contém os valores em V1 em comparação com os valores da linha correspondente da coluna V4. A coluna VX compara as colunas V2 e V5. A coluna VY compara as colunas V3 e V6.

```
TEMP = V1[10:20] != V2 ou TEMP = V1[10:20] != V2[1:11]
```

Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm os resultados da comparação dos valores nas linhas 10-20 da coluna V1 com as linhas 1-11 da coluna V2. Outras células em TEMP estão vazias.

Funções relacionadas

Função	Descrição
EQ	Retornará TRUE se um intervalo de dados for igual ao outro
GE	Retornará TRUE se um intervalo de dados for maior ou igual ao outro
GT	Retornará TRUE se um intervalo de dados for maior que o outro
LE	Retornará TRUE se um intervalo de dados for menor ou igual ao outro
LT	Retornará TRUE se um intervalo de dados for menor que o outro

Macro NOT

O NOT está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
NOT(data) ! data
```

Parâmetros

data

Os valores numéricos dos quais calcular o NOT lógico. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

NOT retorna o NOT lógico dos valores no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo o NOT lógico dos valores na coluna de entrada correspondente. Esta função retorna zero para valores não zero e um para valores zero.

Nota: O operador NOT pode ser abreviado com um ponto de exclamação (!). Use o ponto de exclamação antes do valor dos dados (por exemplo, para especificar NOT(V1), é possível simplesmente digitar (!V1).

Exemplos

```
TEMP = NOT(3.2) ou TEMP = !1
```

Cria uma nova coluna denominada TEMP contendo o valor zero.

`TEMP = !0 ou TEMP = !(2+2=3)` Cria uma nova coluna denominada `TEMP` contendo o valor um.
`TEMP = !V1` Cria uma nova coluna denominada `TEMP`, em que cada valor é o NOT lógico dos valores na coluna `V1`.
`TEMP = !V1:V3` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os NOTs lógicos dos valores na coluna `V1`, os valores da coluna `VX` são os NOTs lógicos dos valores na coluna `V2`, e os valores da coluna `VY` são os NOTs lógicos dos valores na coluna `V3`.
`TEMP = !V1[10:20]` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os NOTs lógicos dos valores nas linhas 10-20 da coluna `V1`. Outras células em `TEMP` estão vazias.
`TEMP = !V1[1:5]:V2` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são os NOTs lógicos dos valores das linhas correspondentes da coluna `V1`, e os valores na coluna `VX` são os NOTs lógicos dos valores das linhas correspondentes da coluna `V2`.

Funções relacionadas

Função	Descrição
`AND`	Calcula o AND lógico entre dois intervalos de dados especificados
`INVERSE`	Calcula o negativo do conteúdo do intervalo de dados especificado
`OR`	Calcula o OR lógico entre dois intervalos de dados especificados
`SIGN`	Calcula o sinal (positivo ou negativo) dos valores no intervalo de dados especificado

Macro NUMBER

A macro NUMBER está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
NUMBER(data [, conversion_keyword])
```

Parâmetros

`data`

Os dados de texto ASCII para converter em valores numéricos. Isso pode ser um texto ASCII entre aspas, uma coluna de texto, um intervalo de células contendo um texto ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`conversion_keyword`

Esta palavra-chave opcional especifica como interpretar os formatos de texto para datas e horas. Selecione uma das palavras-chave na tabela a seguir.

Nota: Se esse parâmetro não for especificado, o padrão será 1.

Palavra-chave de Conversão	Formato	Descrição
0	#####	Converte os 5 primeiros caracteres de cada sequência de texto em um número exclusivo
	$ (default)	Converte valores em dólar para numéricos (por exemplo, "$123,45" para 123,45)
2	%	Converte um valor percentual para numéricos (por exemplo, "50%" para 0.5)
3	mm/dd/yy hh:mm	Converte uma data e hora no número de dias decorridos desde 1º de janeiro de 0000 (1900 é incluído automaticamente no ano yy)
4	dd-mmm-yy	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (1900 é incluído automaticamente no ano yy)
5	mm/dd/yy	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (1900 é incluído automaticamente no ano yy)
6	mmm-yy	Converte uma data no número de dias decorridos entre o primeiro dia do mês especificado e 1º de janeiro de 0000 (1900 é incluído automaticamente no ano yy)
7	dd-mmm	Converte uma data no número de dias desde o início do ano (por exemplo, "01-FEB" para 32)
8	mmm	Converte uma abreviação de mês de 3 letras em um valor entre 1-12 (por exemplo, "DEC" para 12)
9	{January | February | March ... }	Converte um nome de mês escrito por extenso em um valor entre 1-12 (por exemplo, "Março" para 3)
10	{Sun | Mon | Tue ... }	Converte uma abreviação de dia da semana de 3 dias em um valor entre 0-6, em que domingo marca o início da semana (por exemplo, "Sun" para 0)
11	{Sunday | Monday | Tuesday ... }	Converte um nome de dia da semana escrito por extenso em um valor entre 0-6, em que domingo marca o início da semana (por exemplo, "Monday" para 1)
12	hh:mm:ss {AM | PM}	Converte o tempo no número de segundos decorridos desde 00:00:00 AM (meia-noite) (por exemplo, "01:00:00 AM" para 3600)
13	hh:mm:ss	Converte o tempo no número de segundos decorridos desde 00:00:00 AM (meia-noite) (por exemplo, "01:00:00" para 3600)
14	hh:mm {AM | PM}	Converte o tempo no número de minutos decorrido desde 00:00:00 AM (meia-noite) (por exemplo, "01:00 AM" para 60)

Palavra-chave de Conversão	Formato	Descrição
15	hh:mm	Converte o tempo no número de minutos decorrido desde 00:00:00 AM (meia-noite) (por exemplo, "01:00" para 60)
16	mm:ss	Converte o tempo no número de segundos decorridos desde 00:00:00 AM (meia-noite) (por exemplo, "30:00" para 1800)
17	ddmm	Converte uma data no número de dias desde o início do ano (por exemplo, "3101" para 31)
18	ddmmm	Converte uma data no número de dias desde o início do ano (por exemplo, "31JAN" para 31)
19	ddmmmyy	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (1900 será incluído automaticamente no ano se yy for menor ou igual a 20; caso contrário, 2000 será incluído)
20	ddmmmyyyy	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (por exemplo, "31JAN0000" para 31)
21	ddmmyy	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (1900 será incluído automaticamente no ano se yy for menor ou igual a 20; caso contrário, 2000 será incluído)
22	ddmmyyyy	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (por exemplo, "31010000" para 31)
23	mmdd	Converte uma data no número de dias desde o início do ano (por exemplo, "0131" para 31)
24	mmddyy	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (1900 será incluído automaticamente no ano se yy for menor ou igual a 20; caso contrário, 2000 será incluído)
25	mmddyyyy	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (por exemplo, "01010001" para 366)
26	mmm	Converte uma abreviação de mês de 3 letras em um valor entre 1-12 (por exemplo, "MAR" para 3) [Note que este é o mesmo que a palavra-chave de conversão 8]
27	mmmdd	Converte uma data no número de dias desde o início do ano (por exemplo, "JAN31" para 31)
28	mmmddyy	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (1900 será incluído automaticamente no ano se yy for menor ou igual a 20; caso contrário, 2000 será incluído)

Palavra-chave de Conversão	Formato	Descrição
29	mmmddyyyy	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (por exemplo, "FEB010001" para 32)
30	mmmyy	Converte uma data no número de dias decorridos entre o primeiro dia do mês especificado e 1º de janeiro de 0000 (1900 será incluído automaticamente no ano se yy for menor ou igual a 20; caso contrário, 2000 será incluído)
31	mmmyyyy	Converte uma data no número de dias decorridos entre o primeiro dia do mês especificado e 1º de janeiro de 0000 (por exemplo, "FEB0001" para 32)
32	mmyy	Converte uma data no número de dias decorridos entre o primeiro dia do mês especificado e 1º de janeiro de 0000 (1900 será incluído automaticamente no ano se yy for menor ou igual a 20; caso contrário, 2000 será incluído)
33	mmyyyy	Converte uma data no número de dias decorridos entre o primeiro dia do mês especificado e 1º de janeiro de 0000 (por exemplo, "020001" para 32)
34	yymm	Converte uma data no número de dias decorridos entre o primeiro dia do mês especificado e 1º de janeiro de 0000 (1900 será incluído automaticamente no ano se yy for menor ou igual a 20; caso contrário, 2000 será incluído)
35	yymmdd	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (1900 será incluído automaticamente no ano se yy for menor ou igual a 20; caso contrário, 2000 será incluído)
36	yymmm	Converte uma data no número de dias decorridos entre o primeiro dia do mês especificado e 1º de janeiro de 0000 (1900 será incluído automaticamente no ano se yy for menor ou igual a 20; caso contrário, 2000 será incluído)
37	yymmmdd	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (1900 será incluído automaticamente no ano se yy for menor ou igual a 20; caso contrário, 2000 será incluído)
38	yyyy	Converte o ano no número de anos decorridos desde o ano 0000 (por exemplo, "1998" para 1998)
39	yyyymm	Converte uma data no número de dias decorridos entre o primeiro dia do mês especificado e 1º de janeiro de 0000 (por exemplo, "000102" para 32)

Palavra-chave de Conversão	Formato	Descrição
40	yyyymmdd	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (por exemplo, "00010201" para 32)
41	yyyymmm	Converte uma data no número de dias decorridos entre o primeiro dia do mês especificado e 1º de janeiro de 0000 (por exemplo, "000102" para 32)
42	yyyymmmdd	Converte uma data no número de dias decorridos desde 1º de janeiro de 0000 (por exemplo, "0001FEB01" para 32)
43	<day>* <month>	Converte qualquer data delimitada com o dia seguido do mês no número de dias decorridos desde o início do ano (por exemplo, "15-JAN" para 15)
44	<day>* <month>* <year>	Converte qualquer data delimitada com o dia que aparecer antes do mês seguido do ano no número de dias decorridos desde 1º de janeiro de 0000 (por exemplo, "1/1/0001" para 366)
45	<month>* <day>	Converte qualquer data delimitada com o mês seguido do dia no número de dias desde o início do ano (por exemplo, "JAN 31" para 31)
46	<month>* <day>* <year>	Converte qualquer data delimitada com o mês seguido do dia e do ano no número de dias decorridos desde 1º de janeiro de 0000 (por exemplo, "JAN 1, 0001" para 366)
47	<month>* <year>	Converte qualquer data delimitada com mês seguido do ano no número de dias decorridos entre o primeiro dia do mês especificado e 1º de janeiro de 0000
48	<year>* <month>	Converte qualquer data delimitada com ano seguido do mês no número de dias decorridos entre o primeiro dia do mês especificado e 1º de janeiro de 0000
49	<year>* <month>* <day>	Converte qualquer data delimitada com o mês seguido do dia e do ano no número de dias decorridos desde 1º de janeiro de 0000 (por exemplo, "0001/01/01" para 366)
50	yy	Converte o ano no número de anos decorridos desde o ano 0000 (por exemplo, "97" para 97)
51	mm	Converte o mês em um valor entre 1-12 (por exemplo, "SEP" para 9)
52	dd	Converte o dia em um valor entre 1-31 (por exemplo, "28" para 28)
53	{January | February | March ... }	Converte um nome de mês escrito por extenso em um valor entre 1-12 (por exemplo, "Março" para 3) [Note que este é o mesmo que a palavra-chave de conversão 9]

Palavra-chave de Conversão	Formato	Descrição
54	{Sunday | Monday | Tuesday ... }	Converte um nome de dia da semana escrito por extenso em um valor entre 1-7, em que domingo marca o início da semana (por exemplo, "Sunday" para 1)
55	{Sun | Mon | Tue ... }	Converte uma abreviação de dia da semana de 3 dias em um valor entre 1-7, em que domingo marca o início da semana (por exemplo, "Sun" para 1)

Descrição

NUMBER converte valores de texto no intervalo de dados especificado em valores numéricos usando o formato especificado para converter datas e horas. Se uma sequência de texto não puder ser analisada usando a conversion_keyword especificada, NUMBER gerará um erro. O formato 0 converte os cinco primeiros caracteres de cada sequência de texto em um número diferente para cada sequência de texto exclusiva. Esta é uma maneira fácil de alterar uma coluna de texto em classes exclusivas para saídas de um classificador.

Os formatos delimitados (palavras-chave de conversão 43-49) suportam qualquer um dos seguintes como delimitadores:

- / (barra)
- - (traço)
- , (vírgula)
- " " (espaço)
- : (dois pontos)

Os meses podem ser representados como mm ou mmm, os dias podem ser representados como d ou dd, e os anos podem ser representados como yy ou aaaa.

Nota: Em suporte para conformidade do ano 2000, todos os anos nas datas podem ser designados como yyyy ao invés de yy. Para compatibilidade com versões anteriores, palavras-chave de conversão 1-16, yy (anos de 2 dígitos) têm 1900 incluído automaticamente. Para palavras-chave de conversão 17-55, yy < threshold incluiu automaticamente 2000; yy ≥ threshold incluiu automaticamente 1900.

Nota: O valor threshold do ano 2000 é configurado na guia **Limpeza de Dados** da janela **Configurações Avançadas** (denominada usando **Opções > Configurações > Configurações Avançadas**).

Nota: Se alterar o valor do limite do ano 2000, deve-se atualizar todas as funções macro usando a função macro NUMBER para manipular valores de datas com anos de 2 dígitos. Para forçar uma atualização de uma função macro, é possível fazer qualquer edição (por exemplo, incluir um espaço e excluí-lo) e clicar no ícone de marca de seleção para aceitar a mudança.

Nota: Ao utilizar o formato 0, somente os primeiros cinco caracteres de cada sequência de texto são utilizados para gerar um número exclusivo. Todas as sequências com os mesmos cinco primeiros caracteres serão convertidas no mesmo valor numérico. A sequência de texto produzirá o mesmo valor numérico a cada vez, mesmo em planilhas diferentes. Se necessário, utilizar macros de sequência

para manipular sequências de modo que os cinco primeiros caracteres definem exclusivamente uma classe. Observe que os valores numéricos resultantes podem ser muito pequenos. Use a janela **Formatos de Exibição** para aumentar o número de casas decimais exibidas ou alterar o formato para modo exponencial (00E+00).

Exemplos

TEMP = NUMBER("$1.23") ou TEMP = NUMBER("123%", 2) Cria uma nova coluna denominada TEMP contendo o número 1.23.
TEMP = NUMBER(column("Jan", "Mar", "Dec", 8) Cria uma nova coluna denominada TEMP contendo os números 1, 3 e 12.
TEMP = NUMBER("1:52 PM", 14) Cria uma nova coluna denominada TEMP contendo o número 832.
TEMP = NUMBER("1/1/95", 5) Cria uma nova coluna denominada TEMP contendo o número 728660.
TEMP = NUMBER(V1) Cria uma nova coluna denominada TEMP contendo os valores numéricos das sequências de texto na coluna V1. Quaisquer valores em dólar são convertidos corretamente em valores numéricos. ??? 'é retornado para sequências de texto que não puderem ser analisadas usando o formato $.
TEMP = NUMBER(V1:V3, 4) Cria três novas colunas denominadas TEMP, VX e VY. A coluna TEMP contém os valores numéricos de sequências de texto na coluna V1. A coluna VX contém os valores numéricos de sequências de texto na coluna V2. A coluna VY contém os valores numéricos de sequências de texto na coluna V3. Quaisquer datas no formato dd-mmm-aa são convertidas no número de dias de deslocamento a partir de 1º de janeiro de 0000. ??? 'é retornado para sequências de texto que não puderem ser analisadas usando o formato $.
TEMP = NUMBER(V1[10:20]:V2, 10) Cria duas novas colunas denominadas TEMP e VX. A coluna TEMP contém os valores numéricos de sequências de texto nas linhas 10-20 da coluna V1. A coluna VX contém os valores numéricos de sequências de texto nas linhas 10-20 na coluna V2. Todas as três representações de caracteres padrão de dias da semana são convertidas nos números 0-6 (0 = domingo, 6= sábado). Se não houver correspondência para um nome, ??? será retornado.
TEMP = NUMBER(V1, 0) Supondo que a coluna V1 contém todas as sequências de texto de cinco dígitos, cria uma nova coluna denominada TEMP contendo um valor numérico diferente para cada sequência exclusiva.

Funções relacionadas

Função	Descrição
WEEKDAY	Converte sequências de datas de texto ASCII no dia da semana

Macro OR

O OR está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
data1 OR data2 data1 || data2
```

Parâmetros

`data1`

Os números para OR lógico com os valores em `data2`. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`data2`

Um ou mais números para OR lógico com os valores em `data1`. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em `data2` deve ser igual ao número de colunas em `data1`, a menos que `data2` seja uma constante. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`OR` calcula o OR lógico entre os dois intervalos de dados especificados. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em `data1` com OR definido logicamente com a coluna correspondente de `data2` (ou seja, a primeira coluna de `data1` tem o OR definido logicamente na primeira coluna de `data`, a segunda coluna com a segunda coluna, e assim por diante).

Se `data2` for uma constante, cada valor em `data1` terá OR definido logicamente por este valor. Se `data2` contiver uma ou mais colunas, os cálculos serão executados em uma base de linha por linha entre uma coluna `data2` e uma coluna de `data2`. A primeira linha de `data1` tem o OR definido logicamente no valor da primeira linha de `data2`, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: Usar uma coluna contendo o mesmo número *x* em cada linha como `data2` é o mesmo que utilizar a constante *x* como `data2`.

Nota: O operador `OR` pode ser abreviado com uma barra vertical dupla (`||`). Use a barra vertical dupla para separar os dois argumentos (por exemplo, para especificar `V1 OR 3`, é possível simplesmente digitar `V1||3`).

Exemplos

`TEMP = 1 OR 8 ou TEMP = 1 || 8` Cria uma nova coluna denominada `TEMP` que contém o valor um (qualquer número diferente de zero será tratado como um).
`TEMP = V1 || 1` Cria uma nova coluna denominada `TEMP` contendo todos os números um (cada valor de OR executado com o número um produz um).

`TEMP = V1 || V2` Cria uma nova coluna denominada `TEMP`, em que cada valor é o valor da linha da coluna `V1` com OR definido logicamente com o valor da linha correspondente da coluna `V2`.
`TEMP = V1:V3 || V4:V6` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. A coluna `TEMP` contém os valores em `V1` com OR definido logicamente com os valores da linha correspondente da coluna `V4`. A coluna `VX` contém os valores de OR definidos logicamente das colunas `V2` e `V5`. A coluna `VY` contém os valores de OR definidos logicamente das colunas `V3` e `V6`.
`TEMP = V1[10:20] || V2` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm o resultado do OR lógico definido dos valores nas linhas 10-20 das colunas `V1` e `V2`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`AND`	Calcula o AND lógico entre dois intervalos de dados especificados
`NOT`	Calcula o NOT lógico do conteúdo do intervalo de dados especificado

Macro POSITION

O `POSITION` está disponível apenas no IBM Campaign.

Sintaxe

```
POSITION(colName, pattern [, start [, occurrence]])
```

Parâmetros

`colName`

O valor de uma coluna (deve ser um tipo `string`).

`pattern`

O padrão ou a sequência para a qual você está procurando.

`start`

O byte com o qual iniciar a procura.

`occurrence`

Especifique um valor para `n`, em que você está procurando pela `n`-ésima ocorrência do padrão a ser retornada.

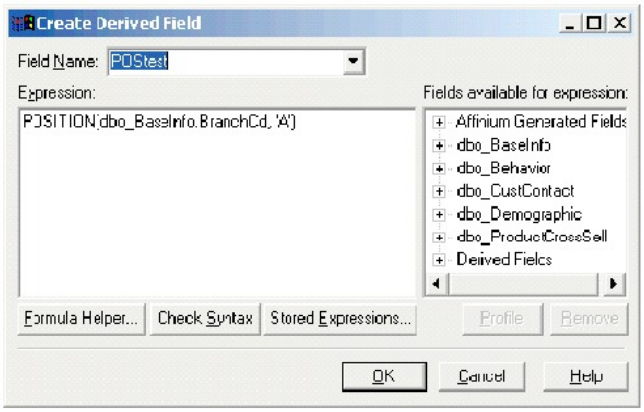
Descrição

`POSITION` retorna a posição inicial do byte de um padrão ou sequência dentro do valor de uma coluna (`colName`) que deve ser um tipo de sequência. Se o início for especificado, ele começará a procurar a partir daí. A ocorrência é a enésima ocorrência de padrão a ser retornada.

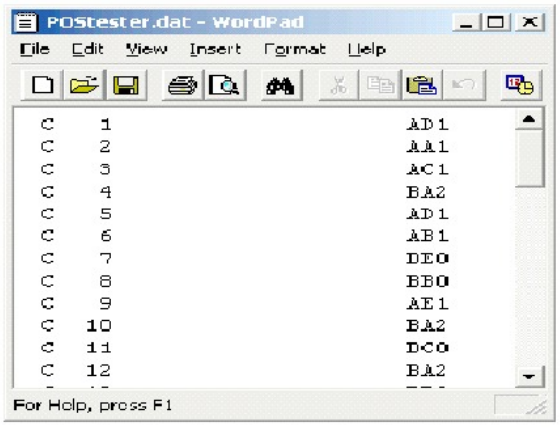
Nota: A procura não faz distinção entre maiúsculas e minúsculas.

Exemplos

No exemplo abaixo, estamos procurando pelo padrão ou pela sequência `'A'`, dentro do valor da coluna `dbo_BaseInfo.BranchCd`, e designando o valor retornado para um arquivado `POStest` derivado.



O exemplo a seguir mostra algumas linhas da tabela com os valores de `dbo_BaseInfo.BranchCd` e de `POStest` mostrados lado a lado.



Um exemplo mais complexo:

```
STRING_SEG(POSITION(CellCode,"X",1,2)+1,

STRING_LENGTH(CellCode),CellCode) = "AAA"
```

Isso retorna linhas em que os valores de CellCode possuem "AAA " no final após a segunda ocorrência de "X ".

Macro PLUS

O PLUS está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
data PLUS addend data + addend
```

Parâmetros

data

O intervalo de células contendo números para somar. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

addend

Um ou mais números para somar a todos os valores na coluna especificada. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em data2 deve ser igual ao número de colunas em data1, a menos que data2 seja uma constante. Para obter a definição de formato de addend (mesmo que data), consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

PLUS soma os valores nos dois intervalos de dados especificados. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em data1 somado com a coluna correspondente de data2 (ou seja, a primeira coluna de data1 é somada com a primeira coluna de data, a segunda coluna com a segunda coluna, e assim por diante).

Se data2 for uma constante, cada valor em data1 será aumentado por este valor. Se data2 contiver uma ou mais colunas, os cálculos serão executados em uma base de linha por linha entre uma coluna data2 e uma coluna de data2. A primeira linha de data1 é somada ao valor da primeira linha de data2, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: O operador PLUS pode ser abreviado com um sinal de mais (+).

Exemplos

TEMP = 3 PLUS 4 ou TEMP = 3 + 4 Cria uma nova coluna denominada TEMP contendo o valor sete.
TEMP = V1 + 8 Cria uma nova coluna denominada TEMP, em que cada valor é o conteúdo da coluna V1 mais oito.

`TEMP = V1 + V1` Cria uma nova coluna denominada `TEMP` contendo duas vezes o conteúdo da coluna `V1`.
`TEMP = V1 + V2` Cria uma nova coluna denominada `TEMP`, em que cada valor é o valor da linha da coluna `V1` mais o valor da linha correspondente da coluna `V2`.
`TEMP = V1:V3 + V4:V6` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. A coluna `TEMP` contém os valores em `V1` mais os valores da linha correspondente da coluna `V4`. A coluna `VX` soma as colunas `V2` de `V5`. A coluna `VY` soma as colunas `V5` e `V6`.
`TEMP = V1[10:20] + V2` ou `TEMP = V1[10:20] + V2[1:11]` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm as somas dos valores nas linhas 10-20 da coluna `v1` e dos valores nas linhas 1-11 da coluna `V2`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`MINUS`	Subtrai um intervalo de dados especificado de outro
`SUM` ou `TOTAL`	Calcula a soma de um intervalo de células

Macro POW

O `POW` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
base POW exponent base ^ exponent
```

Parâmetros

`base`

Os valores numéricos para elevar a uma potência exponencial. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `base` (mesmo que `data`), consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`exponent`

Um ou mais números exponenciais pelos quais elevar os valores em `data`. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em `exponent` deve ser igual ao número de colunas em `base`, a menos que `base` seja uma constante. Para obter a definição de formato de `exponent` (mesmo que `data`), consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

POW eleva os valores no primeiro intervalo de dados à potência especificada no segundo intervalo de dados (ou seja, calcula $base^{exponent}$). Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo o resultado de elevar o base à potência exponent (ou seja, a primeira coluna de data1 é elevada à primeira coluna de data, a segunda coluna com a segunda coluna, e assim por diante).

Se exponent for uma constante, cada valor em base será elevado por este valor. Se exponent contiver uma ou mais colunas, os cálculos serão executados em uma base de linha por linha entre uma coluna base e uma coluna de exponent. A primeira linha de base é elevada ao valor da primeira linha do exponent, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: O operador POW pode ser abreviado com um circunflexo (^). Por exemplo, TEMP = 2^8 é equivalente a TEMP = 2 POW 8.

Nota: Se o valor *x* for muito grande ou muito pequeno, um estouro será retornado. Isso ocorrerá se base^exponent exceder o valor de vírgula flutuante máximo ou mínimo de 32 bits.

Exemplos

TEMP = 2 POW 3 ou TEMP = 2^3 Cria uma nova coluna denominada TEMP contendo o valor oito.
TEMP = V1 ^ 0.5 Cria uma nova coluna denominada TEMP, em que cada valor é a raiz quadrada do conteúdo da coluna V1 (isto é equivalente a SQRT(V1)).
TEMP = V1 ^ V3 Cria uma nova coluna denominada TEMP, em que cada valor é o valor da linha da coluna V1 elevado ao valor da linha correspondente da coluna V2.
TEMP = V1:V3 ^ V4:V6 Cria três novas colunas denominadas TEMP, VX e VY. A coluna TEMP contém os valores em V1 elevado aos valores da linha correspondente da coluna V4. A coluna VX contém o resultado de coluna V2 elevado aos valores correspondentes na coluna V5. A coluna VY contém o resultado da coluna V3 elevado aos valores correspondentes da V6.
TEMP = V1[10:20] POW V2 ou TEMP = V1[10:20] POW V2[1:11] Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm o resultado de elevar os valores nas linhas 10-20 da coluna V1 pelos valores nas linhas 1-10 da coluna V2. Outras células em TEMP estão vazias.

Funções relacionadas

Função	Descrição
EXP	Calcula um ou mais números naturais elevados ao conteúdo de cada célula no intervalo de dados especificado
LN ou LOG	Calcula o log natural do conteúdo do intervalo de dados especificado

Função	Descrição
LN2	Calcula o log de base2 do conteúdo do intervalo de dados especificado
LN10	Calcula o log de base10 do conteúdo do intervalo de dados especificado

Macro RANDOM

A está disponível apenas no IBM Campaign. RANDOM

Sintaxe

```
RANDOM(num [, seed]) RANDOM(num, value1, value2 [, seed])
```

Parâmetros

num

A quantia de números aleatórios para gerar. Esse valor deve ser um número inteiro positivo maior que zero.

value1

Um limite nos números aleatórios para gerar. Isso pode ser qualquer valor constante ou uma expressão avaliada para uma constante. Se esse parâmetro não for fornecido, o padrão será zero.

value2

O outro limite sobre os números aleatórios para gerar. Isso pode ser qualquer valor constante ou uma expressão avaliada para uma constante. Se esse parâmetro não for fornecido, o padrão será um.

seed

Um valor inicial opcional a ser usado para a geração de número aleatório. Esse deve ser um número inteiro.

Descrição

RANDOM gera uma coluna de números aleatórios. Ele retorna uma nova coluna contendo num números aleatórios. Se value1 e value2 forem especificados, os números aleatórios serão gerados entre (e incluindo) estes limites. Se eles não forem especificados, o padrão será gerar valores entre zero e um. Se seed for fornecido, ele será usado como um valor inicial para o gerador de número aleatório.

Nota: Se o seed for maior ou igual a 2^{32} , o valor será substituído por $2^{32} - 1$.

Valores de seed acima de 2^{24} serão arredondados (isto é, a precisão será perdida). Portanto, valores múltiplos podem resultar no mesmo valor de seed.

Exemplos

`TEMP = RANDOM()` Cria uma nova coluna denominada `TEMP` contendo números aleatórios de comprimento ilimitado.
`TEMP = RANDOM(100)` Cria uma nova coluna denominada `TEMP` contendo 100 números aleatórios entre 0,0 e 1,0.
`TEMP = RANDOM(100, 5943049)` Cria uma nova coluna denominada `TEMP` contendo 100 números aleatórios gerados a partir do número de valor inicial 5943049.
`TEMP = RANDOM(100, 0, 100)` Cria uma nova coluna denominada `TEMP` contendo 100 números aleatórios entre 0 e 100,0.
`TEMP = RANDOM(100, 0, 100, 5943049)` Cria uma nova coluna denominada `TEMP` contendo 100 números aleatórios entre -0 e 100 gerados a partir do número de valor inicial 5943049.

Funções Relacionadas

Função	Descrição
`RANDOM_GAUSS`	Retorna o número especificado de valores aleatórios a partir de uma distribuição gaussiana

Macro RANDOM_GAUSS

O `RANDOM_GAUSS` está disponível apenas no IBM Campaign.

Sintaxe

```
RANDOM_GAUSS(num [, seed]) RANDOM_GAUSS(num, mean, std [, seed])
```

Parâmetros

`num`

A quantia de números aleatórios para gerar. Esse valor deve ser um número inteiro positivo maior que zero.

`mean`

A média do Gaussiano. Isso pode ser qualquer valor constante ou uma expressão avaliada para uma constante. Se esse parâmetro não for fornecido, o padrão será zero.

`std`

O desvio padrão do Gaussiano. Isso pode ser qualquer valor constante ou uma expressão avaliada para uma constante. Se esse parâmetro não for fornecido, o padrão será um.

seed

Um valor inicial opcional a ser usado para a geração de número aleatório. Esse deve ser um número inteiro.(Se um valor de não número inteiro for fornecido, o piso do valor será usado automaticamente).

Descrição

RANDOM_GAUSS gera uma coluna com números aleatórios com base em uma distribuição gaussiana. Ele retorna uma nova coluna contendo num números aleatórios. Se mean e std forem especificados, os números aleatórios serão gerados usando uma distribuição gaussiana com os desvios médio e padrão especificados. Se eles não forem especificados, o Gaussiano padrão possuirá uma média de zero e o desvio padrão de um. Se seed for fornecido, ele será usado como um valor inicial para o gerador de número aleatório.

Exemplos

`TEMP = RANDOM_GAUSS(100)` Cria uma nova coluna denominada TEMP contendo 100 valores amostrados aleatoriamente a partir de um Gaussiano de desvio padrão de unidade de média zero.
`TEMP = RANDOM_GAUSS(500, 3)` Cria uma nova coluna denominada TEMP contendo 100 valores amostrados aleatoriamente a partir de um Gaussiano de desvio padrão de unidade de média zero. O número 3 é utilizado como um valor inicial para o gerador de número aleatório.
`TEMP = RANDOM_GAUSS(5000, 100, 32)` Cria uma nova coluna denominada TEMP contendo 5000 valores amostrados aleatoriamente a partir de um Gaussiano com uma média de 100 e desvio padrão de 32.
`TEMP = RANDOM_GAUSS(500, -1, 2, 3)` Cria uma nova coluna denominada TEMP contendo 500 valores amostrados aleatoriamente a partir de um Gaussiano com uma média de -1 e desvio padrão de 2. O número 3 é usado como um valor inicial para o gerador de número aleatório.

Funções relacionadas

Função	Descrição
RANDOM	Retorna o número especificado de números aleatórios

Macro ROUND

O ROUND está disponível apenas no IBM Campaign.

Sintaxe

```
ROUND(data)
```

Parâmetros

data

Os valores numéricos para arredondar. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`ROUND` arredonda os valores no intervalo de dados especificado para o número inteiro mais próximo. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo o valor arredondado de números na coluna de entrada correspondente. Os números que estiverem exatamente na metade são arredondados para cima, (por exemplo, `2.5` é arredondado para `3.0` e `2.5` é arredondado para `-2.0`).

Exemplos

`TEMP = ROUND(3.2)` Cria uma nova coluna denominada `TEMP` contendo o valor três.
`TEMP = ROUND(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é o valor arredondado do conteúdo da coluna `V1`.
`TEMP = ROUND(V1:V3)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os valores arredondados do conteúdo da coluna `V1`, os valores da coluna `VX` são os valores arredondados do conteúdo da coluna `V2`, e os valores da coluna `VY` são os valores arredondados do conteúdo da coluna `V3`.
`TEMP = ROUND(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm os valores arredondados nas linhas 10-20 da coluna `V1`. Outras células em `TEMP` estão vazias.
`TEMP = ROUND(V1[1:5]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são os valores arredondados das linhas correspondentes da coluna `V1`, e os valores na coluna `VX` são os valores arredondados das linhas correspondentes da coluna `V2`.

Funções relacionadas

Função	Descrição
`INT`	Calcula o valor de número inteiro (arredondado para baixo) do conteúdo do intervalo de dados especificado
`MOD`	Calcula o módulo do conteúdo do intervalo de dados especificado
`TRUNCATE`	Retorna a parte não fracionária de cada valor no intervalo de dados especificado.

Macro ROWNUM

O `ROWNUM` está disponível apenas no IBM Campaign.

Sintaxe

```
ROWNUM()
```

Descrição

ROWNUM gera números sequenciais de um até o número de registros. O número para o primeiro registro é um, dois para o segundo registro, e assim por diante

Nota: O número máximo de registros que o ROWNUM pode manipular é dois bilhões.

Macro RTRIM

A macro RTRIM está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
RTRIM(data)
```

Parâmetros

data

Descrição

RTRIM remove caracteres de espaços à direita de cada valor de sequência no intervalo de dados especificado, retornando a sequência convertida. Ele retorna uma nova coluna para cada coluna de entrada.

Exemplos

```
Temp = RTRIM "gold "
```

Cria uma nova sequência denominada Temp que contém "gold".

Macro SIGN

O SIGN está disponível apenas no IBM Campaign.

Sintaxe

```
SIGN(data)
```

Parâmetros

data

Os valores numéricos dos quais calcular o sinal. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

SIGN testa o sinal dos valores no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo o sinal dos números na coluna de entrada correspondente. Um positivo é retornado para todos os valores maiores que zero; um negativo é retornado para todos os valores menores que zero; e zero é retornado para valores de zero.

Exemplos

`TEMP = SIGN(-3)` Cria uma nova coluna denominada TEMP contendo o valor -1.
`TEMP = SIGN(V1)` Cria uma nova coluna denominada TEMP, em que cada valor é o sinal do conteúdo da coluna V1.
`TEMP = SIGN(V1:V3)` Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os sinais do conteúdo da coluna V1, os valores da coluna VX são os sinais do conteúdo da coluna V2, e os valores da coluna VY são os sinais do conteúdo da coluna V3.
`TEMP = SIGN(V1[10:20])` Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm os sinais dos valores nas linhas 10-20 da coluna V1. Outras células em TEMP estão vazias.
`TEMP = SIGN(V1[10:50]:V2)` Cria duas novas colunas denominadas TEMP e VX, cada uma com valores nas linhas 1-41 (as outras células estão vazias). Os valores na coluna TEMP são os sinais dos valores nas linhas 10-50 da coluna V1, e os valores na coluna VX são os sinais dos valores nas linhas 10-50 da coluna V2.

Funções Relacionadas

Função	Descrição
ABS	Calcula o valor absoluto do conteúdo do intervalo de dados especificado
INVERSE	Calcula o negativo do conteúdo do intervalo de dados especificado

Macro SIN

O SIN está disponível apenas no IBM Campaign.

Sintaxe

```
SIN(data [, units_keyword])
```

Parâmetros

data

Os valores numéricos dos quais calcular o seno. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer

item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

units_keyword

Essa palavra-chave opcional determina se os valores de entrada e os resultados são interpretados como graus ou radianos. Selecione uma das opções a seguir:

RADIAN - Executa os cálculos em radianos (padrão)

DEGREE - Executa o cálculo em graus

Se esse parâmetro não for especificado, o padrão será radianos. (Para converter de radianos em graus, divida pelo PI e multiplique por 180).

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte “Especificações de Formato” na página 9.

Descrição

SIN calcula o seno dos valores no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo o seno dos números na coluna de entrada correspondente.

Exemplos

TEMP = SIN(PI/2) ou TEMP = SIN(PI/2, 0) ou TEMP = SIGN(PI/2, RADIAN) Cria uma nova coluna denominada TEMP contendo o valor um.
TEMP = SIN(V1) Cria uma nova coluna denominada TEMP, em que cada valor é o seno (em radianos) do conteúdo da coluna V1.
TEMP = SIN(V1:V3, 1) ou TEMP = SIN(V1:V3, DEGREE) Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os senos do conteúdo da coluna V1, os valores da coluna VX são os senos do conteúdo da coluna V2, e os valores da coluna VY são os senos do conteúdo da coluna V3. Todos os valores estão em graus.
TEMP = SIN(V1[10:50]:V2) Cria duas novas colunas denominadas TEMP e VX, cada uma com valores nas linhas 1-41 (as outras células estão vazias). Os valores na coluna TEMP são os senos dos valores nas linhas 10-50 da coluna V1, e os valores na coluna VX são os senos dos valores nas linhas 10-50 da coluna V2. Todos os valores estão em radianos.

Funções relacionadas

Função	Descrição
ASIN	Calcula o arco seno do conteúdo do intervalo de dados especificado
COS	Calcula o cosseno do conteúdo do intervalo de dados especificado
SINH	Calcula o seno hiperbólico do conteúdo do intervalo de dados especificado

Função	Descrição
TAN	Calcula a tangente do conteúdo do intervalo de dados especificado

Macro SINH

O SINH está disponível apenas no IBM Campaign.

Sintaxe

```
SINH(data [, units_keyword])
```

Parâmetros

data

Os valores numéricos dos quais calcular o seno hiperbólico. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

units_keyword

Essa palavra-chave opcional determina se os valores de entrada e os resultados são interpretados como graus ou radianos. Selecione uma das opções a seguir:

RADIAN - Executa os cálculos em radianos (padrão)

DEGREE - Executa o cálculo em graus

Se esse parâmetro não for especificado, o padrão será radianos. (Para converter de radianos em graus, divida pelo PI e multiplique por 180).

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte “Especificações de Formato” na página 9.

Descrição

SINH calcula o seno hiperbólico dos valores no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo o seno hiperbólico dos números na coluna de entrada correspondente. Para o *x* em radianos, o seno hiperbólico de um número é:

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

em que *e* é o número natural, 2,7182818.

Nota: Se o valor de *x* for muito grande, um erro de estouro será retornado. Isso ocorrerá se o *sinh* (*x*) exceder o valor de vírgula flutuante máximo de 32 bits.

Exemplos

`TEMP = SINH(1)` ou `TEMP = SINH(1, 0)` ou `TEMP = SINH(1, RADIAN)` Cria uma nova coluna denominada `TEMP` contendo o valor `1.18`.
`TEMP = SINH(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é o seno hiperbólico (em radianos) do conteúdo da coluna `V1`.
`TEMP = SINH(V1:V3, 1)` ou `TEMP = SINH(V1:V3, DEGREE)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os senos hiperbólicos do conteúdo da coluna `V1`, os valores da coluna `VX` são os senos hiperbólicos do conteúdo da coluna `V2`, e os valores da coluna `VY` são os senos hiperbólicos do conteúdo da coluna `V3`. Todos os valores estão em graus.
`TEMP = SINH(V1[10:50]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-41 (as outras células estão vazias). Os valores na coluna `TEMP` são os senos hiperbólicos dos valores nas linhas 10-50 da coluna `V1`, e os valores na coluna `VX` são os senos hiperbólicos dos valores nas linhas 10-50 da coluna `V2`. Todos os valores estão em radianos.

Funções relacionadas

Função	Descrição
`COSH`	Calcula o cosseno hiperbólico do conteúdo do intervalo de dados especificado
`SIN`	Calcula o seno do conteúdo do intervalo de dados especificado
`TANH`	Calcula a tangente hiperbólica do conteúdo do intervalo de dados especificado

Macro SQRT

O `SQRT` está disponível apenas no IBM Campaign.

Sintaxe

`SQRT(data)`

Parâmetros

`data`

Os valores numéricos dos quais calcular a raiz quadrada. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`SQRT` calcula a raiz quadrada dos valores no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a raiz quadrada positiva dos números na coluna de entrada correspondente.

Nota: Se um valor no intervalo de dados definido for negativo, um ??? será retornado para essa célula.

Exemplos

`TEMP = SQRT(2)` Cria uma nova coluna denominada `TEMP` contendo o valor `1.41`.
`TEMP = SQRT(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é a raiz quadrada do conteúdo da coluna `V1`.
`TEMP = SQRT(V1:V3)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são as raízes quadradas do conteúdo da coluna `V1`, os valores da coluna `VX` são as raízes quadradas do conteúdo da coluna `V2`, e os valores da coluna `VY` são as raízes quadradas do conteúdo da coluna `V3`.
`TEMP = SQRT(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm as raízes quadradas dos valores nas linhas 10-20 da coluna `V1`. Outras células em `TEMP` estão vazias.
`TEMP = SQRT(V1[10:50]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-41 (as outras células estão vazias). Os valores na coluna `TEMP` são as raízes quadrada dos valores nas linhas 10-50 da coluna `V1`, e os valores na coluna `VX` são as raízes quadrada dos valores nas linhas 10-50 da coluna `V2`.

Funções relacionadas

Função	Descrição
`DIV`	Divide um intervalo de dados especificado por outro
`MULT`	Multiplica o conteúdo de dois intervalos de dados
`POW`	Calcula um valor base elevado a uma ou mais potências exponenciais especificadas

Macro STDV ou STDEV

O `STDV` ou `STDEV` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

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

Parâmetros

`data`

Os valores numéricos dos quais calcular o desvio padrão. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

keyword

Essa palavra-chave opcional determina como o cálculo será executado sobre o intervalo de dados de entrada. Selecione uma das opções a seguir:

ALL - Executa o cálculo em todas as células em data (padrão)

COL - Executa o cálculo separadamente para cada coluna de data

ROW - Executa o cálculo separadamente para cada linha de data

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte "Especificações de Formato" na página 9.

Nota: Muitas funções de macro utilizam os parâmetros de palavra-chave {ALL | COL | ROW}. Estas palavras-chave não se aplicam ao **IBM Campaign** porque os dados de entrada são sempre uma coluna ou um campo único. A macro sempre se comportará como se a palavra-chave COL fosse especificada. Portanto, não é necessário especificar estas palavras-chave quando usar **IBM Campaign** .

Descrição

STDV calcula o desvio padrão de todas as células no intervalo de dados especificado. O desvio padrão de uma distribuição é a raiz quadrada da variância. O desvio padrão é calculado conforme a seguir:

$$\sqrt{\frac{1}{n-1}\sum_{j=1}^{n}(x_j - \text{mean})^2}$$

em que *x* são as amostras, *n* é o número de amostras e *mean* é a média da distribuição.

Nota: Se o número de amostras for *n* = 1, STDV retornará um erro.

Exemplos

TEMP = STDV(V1) Cria uma nova coluna denominada TEMP contendo um valor único que é o desvio padrão do conteúdo da coluna V1.
TEMP = STDV(V1:V3) Cria uma nova coluna denominada TEMP contendo um valor único que é o desvio padrão do conteúdo das colunas V1, V2 e V3.
TEMP = STDV(V1[1:5]:V4) Cria uma nova coluna denominada TEMP contendo um valor único que é o desvio padrão das células nas linhas 1-5 das colunas V1 a V4.

`TEMP = STDV(V1:V3, COL)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. O valor único na coluna `TEMP` é o desvio padrão do conteúdo da coluna `V1`, o valor único na coluna `VX` é o desvio padrão do conteúdo da coluna `V2`, e o valor único na coluna `VY` é o desvio padrão do conteúdo da coluna `V3`.
`TEMP = STDV(V1[10:50]:V3,COL)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`, cada uma contendo um valor único. O valor na coluna `TEMP` é o desvio padrão das células nas linhas 10-50 da coluna `V1`, o valor na coluna `VX` é o desvio padrão das células nas linhas 10-50 da coluna `V2`, e o valor na coluna `VY` é o desvio padrão das células nas linhas 10-50 da coluna `V3`.
`TEMP = STDV(V1:V3, ROW)` Cria uma nova coluna denominada `TEMP`, em que cada entrada na célula é o desvio padrão da linha correspondente nas colunas `V1`, `V2` e `V3`.
`TEMP = STDV(V1[1:5]:V3,ROW)` Cria uma nova coluna denominada `TEMP`, em que as células nas linhas 1-5 contêm os desvios padrão da linha correspondente nas colunas `V1` a `V3`. Outras células em `TEMP` estão vazias.

Funções Relacionadas

Função	Descrição
`VAR`	Calcula a variância de um intervalo de células

Macro STRING_CONCAT

O `STRING_CONCAT` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
STRING_CONCAT(string1, string2, ... stringN)
```

Parâmetros

`string`

Uma sequência de texto ASCII para concatenar. Isso pode ser um texto ASCII entre aspas, uma coluna de texto, um intervalo de células contendo um texto ou uma expressão avaliada para qualquer item acima. Consulte a seção *Parâmetros de Função Macro* do capítulo *Utilizando Macros* para seu produto para obter a definição de formato de `string` (o mesmo que `data`).

Descrição

`STRING_CONCAT` concatena os valores de texto ASCII nos intervalos de dados especificados. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo as sequências concatenadas das linhas correspondentes de `strings`. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: A largura total de cada sequência resultante não pode exceder 255 caracteres.

O IBM Interact também suporta a seguinte sintaxe:

`STRING_CONCAT(` *string1* `,` *string2* `, ...` *stringN* `)`

Por exemplo, `STRING_CONCAT('a', 'b', 'c', 'd')` é válida.

Exemplos

`TEMP = STRING_CONCAT("house", "boat")` Cria uma nova coluna denominada `TEMP`, que contém a sequência de texto ASCII "`houseboat`".
`TEMP = STRING_CONCAT(V1, ".")` Cria uma nova coluna denominada `TEMP`, com cada linha contendo a sequência de texto ASCII na linha correspondente da coluna `V1` com um ponto anexado.
`TEMP = STRING_CONCAT(V1, V2)` Cria uma nova coluna denominada `TEMP`, com cada linha contendo a sequência de texto ASCII na coluna `V1` concatenada com a sequência de texto na coluna `V2`.
`TEMP = STRING_CONCAT(V1:V3, V4:V6)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são as sequências concatenadas das linhas correspondentes das colunas `V1` e `V4`, os valores da coluna `VX` são as sequências concatenadas das linhas correspondentes das colunas `V2` e `V5`, e os valores da coluna `VY` são as sequências concatenadas das linhas correspondentes das colunas `V3` e `V6`.
`TEMP = STRING_CONCAT(V1[5:10]:V2, V3:V4)` Cria duas novas colunas denominadas `TEMP` e `VX`. Os valores na coluna `TEMP` são sequências das linhas 5-10 da coluna `V1` concatenadas com as linhas 1-6 da coluna `V3`. Os valores em `VX` são as sequências das linhas 5-10 da coluna `V2` concatenadas com as linhas 1-6 da coluna `V4`.
`TEMP = STRING_CONCAT('a', 'b', 'c', 'd')` Cria uma nova coluna denominada `TEMP`, que contém a sequência de texto ASCII "`abcd`".

Funções Relacionadas

Função	Descrição
`STRING_HEAD`	Retorna os primeiros n caracteres de cada sequência no intervalo de dados especificado
`STRING_LENGTH`	Retorna o comprimento de cada sequência no intervalo de dados especificado.
`STRING_SEG`	Retorna o segmento da sequência entre dois índices especificados
`STRING_TAIL`	Retorna os últimos n caracteres de cada sequência no intervalo de dados especificado.

Macro STRING_HEAD

O STRING_HEAD está disponível apenas no IBM Campaign.

Sintaxe

```
STRING_HEAD(num_chars, data)
```

Parâmetros

num_chars

O número de caracteres para retornar a partir do início de cada sequência em data. Este deve ser um número inteiro positivo maior que zero.

data

Valores de sequência de texto ASCII. Isso pode ser um texto ASCII entre aspas, uma coluna de texto, um intervalo de células contendo um texto ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

STRING_HEAD retorna os primeiros num_chars caracteres de cada valor de sequência no intervalo de dados especificado. Se num_chars for maior que o número de caracteres em uma sequência de texto, os caracteres restantes serão preenchidos com o caractere nulo "\0".

Exemplos

`TEMP = STRING_HEAD(3, "JAN 15, 1997")` Cria uma nova coluna denominada TEMP, que contém a sequência de texto ASCII "JAN".
`TEMP = STRING_HEAD(10, "Pressure")` Cria uma nova coluna denominada TEMP, que contém a sequência de texto ASCII "Pressure".
`TEMP = STRING_HEAD(5, V1)` Cria uma nova coluna denominada TEMP contendo os cinco primeiros caracteres de cada sequência na coluna V1.
`TEMP = STRING_HEAD(1, V1:V3)` Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os primeiros caracteres das sequências nas linhas correspondentes da coluna V1, os valores da coluna VX são os primeiros caracteres das sequências nas linhas correspondentes da coluna V2, e os valores da coluna VY são os primeiros caracteres das sequências nas linhas correspondentes de coluna V3.
`TEMP = STRING_HEAD(12, V4[1:50]:V6]` Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os 12 primeiros caracteres das sequências nas linhas 1-50 da coluna V1, os valores da coluna VX são os 12 primeiros caracteres das sequências nas linhas 1-50 da coluna V2, e os valores da coluna VY são os 12 primeiros caracteres das sequências nas linhas 1-50 da coluna V3.

Funções relacionadas

Função	Descrição
STRING_CONCAT	Concatena duas sequências de texto a partir dos intervalos de dados especificados
STRING_LENGTH	Retorna o comprimento de cada sequência no intervalo de dados especificado.
STRING_SEG	Retorna o segmento da sequência entre dois índices especificados
STRING_TAIL	Retorna os últimos n caracteres de cada sequência no intervalo de dados especificado.

Macro STRING_LENGTH

O STRING_LENGTH está disponível apenas no IBM Campaign.

Sintaxe

```
STRING_LENGTH(data)
```

Parâmetros

data

Os valores de sequência de texto ASCII dos quais calcular o comprimento. Isso pode ser um texto ASCII entre aspas, uma coluna de texto, um intervalo de células contendo um texto ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

STRING_LENGTH retorna o comprimento de cada valor de sequência no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo o comprimento da sequência de texto correspondente.

Nota: Se STRING_LENGTH for aplicado a colunas contendo dados numéricos, ele retornará zeros.

Exemplos

`TEMP = STRING_LENGTH("four")` Cria uma nova coluna denominada TEMP contendo o valor 4.
`TEMP = STRING_LENGTH(4)` Cria uma nova coluna denominada TEMP contendo o valor 0.
`TEMP = STRING_LENGTH(V1)` Cria uma nova coluna denominada TEMP, em que cada valor é o comprimento da sequência na linha correspondente da coluna V1.

`TEMP = STRING_LENGTH(V1:V3)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os comprimentos das sequências nas linhas correspondentes da coluna `V1`, os valores da coluna `VX` são os comprimentos das sequências nas linhas correspondentes da coluna `V2`, e os valores da coluna `VY` são os comprimentos das sequências nas linhas correspondentes de coluna `V3`.
`TEMP = STRING_LENGTH(V4[1:50]:V6]` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os comprimentos das sequências nas linhas 1-50 da coluna `V1`, os valores da coluna `VX` são os comprimentos das sequências nas linhas 1-50 da coluna `V2`, e os valores da coluna `VY` são os comprimentos das sequências nas linhas 1-50 da coluna `V3`.

Funções relacionadas

Função	Descrição
`STRING_CONCAT`	Concatena duas sequências de texto a partir dos intervalos de dados especificados
`STRING_HEAD`	Retorna os primeiros n caracteres de cada sequência no intervalo de dados especificado
`STRING_SEG`	Retorna o segmento da sequência entre dois índices especificados
`STRING_TAIL`	Retorna os últimos n caracteres de cada sequência no intervalo de dados especificado.

Macro STRING_PROPER

O `STRING_PROPER` está disponível apenas no IBM Campaign.

Sintaxe

```
STRING_PROPER(data)
```

Parâmetros

`data`

O valor de sequência a ser convertido.

Descrição

`STRING_PROPER` converte cada valor de sequência no intervalo de dados especificado ao alterar a primeira letra ou qualquer letra após um caractere de espaço em branco ou símbolo (diferente de sublinhado) em letras maiúsculas, e todos os outros caracteres em letras minúsculas. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a sequência convertida na coluna de entrada correspondente.

Exemplos

```
Temp = STRING_PROPER
```

Macro STRING_SEG

O STRING_SEG está disponível apenas no IBM Campaign.

Sintaxe

```
STRING_SEG(from, to, data)
```

Parâmetros

de

O número de deslocamento de caracteres do início da sequência até o início da extração do segmento de sequência from. Este deve ser um número inteiro positivo maior que zero e menor que to, ou STRING_SEG retornará uma sequência vazia.

to

O número de deslocamento de caracteres do início da sequência até parar a extração do segmento de sequência from. Este deve ser um número inteiro positivo maior ou igual a from. Se to for igual a from (e to for menor ou igual ao comprimento da sequência), um caractere será retornado.

data

Valores de sequência de texto ASCII. Isso pode ser um texto ASCII entre aspas, uma coluna de texto, um intervalo de células contendo um texto ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

STRING_SEG retorna o segmento de sequência entre dois índices de cada valor de sequência no intervalo de dados especificado. Se from for maior que o comprimento de uma sequência, nada será retornado. Se to for maior que o comprimento de uma sequência, todos os caracteres de from serão retornados.

Exemplos

`TEMP = STRING_SEG(1, 6, "JAN 15, 1997")` Cria uma nova coluna denominada TEMP, que contém a sequência de texto ASCII "Jan 15".
`TEMP = STRING_SEG(5, 20, "Pressure")` Cria uma nova coluna denominada TEMP, que contém a sequência de texto ASCII "sure".
`TEMP = STRING_SEG(5, 6, V1)` Cria uma nova coluna denominada TEMP contendo o quinto e o sexto caracteres de cada sequência na coluna V1.

`TEMP = STRING_SEG(10, 20, V1:V3)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os caracteres 10-20 das sequências nas linhas correspondentes da coluna `V1`, os valores da coluna `VX` são os caracteres 10-20 das sequências nas linhas correspondentes da coluna `V2`, e os valores da coluna `VY` são os caracteres 10-20 das sequências nas linhas correspondentes de coluna `V3`.
`TEMP = STRING_SEG(5, 10, V4[1:50]:V6]` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são os caracteres 5-10 das sequências nas linhas 1-50 da coluna `V1`, os valores da coluna `VX` são os caracteres 5-10 das sequências nas linhas 1-50 da coluna `V2`, e os valores da coluna `VY` são os caracteres 5-10 das sequências nas linhas 1-50 da coluna `V3`.

Funções relacionadas

Função	Descrição
`STRING_CONCAT`	Concatena duas sequências de texto a partir dos intervalos de dados especificados
`STRING_HEAD`	Retorna os primeiros n caracteres de cada sequência no intervalo de dados especificado
`STRING_LENGTH`	Retorna o comprimento de cada sequência no intervalo de dados especificado.
`STRING_TAIL`	Retorna os últimos n caracteres de cada sequência no intervalo de dados especificado.

Macro STRING_TAIL

O `STRING_TAIL` está disponível apenas no IBM Campaign.

Sintaxe

```
STRING_TAIL(num_chars, data)
```

Parâmetros

`num_chars`

O número de caracteres para retornar a partir do término de cada sequência em `data`. Este deve ser um número inteiro positivo maior que zero.

`data`

Valores de sequência de texto ASCII. Isso pode ser um texto ASCII entre aspas, uma coluna de texto, um intervalo de células contendo um texto ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`STRING_TAIL` retorna os últimos `num_chars` caracteres de cada valor de sequência no intervalo de dados especificado. Todos os valores de sequência são preenchidos até

o comprimento da sequência mais longa com caracteres nulos "\0". Em seguida, os últimos num_chars são retornados de cada sequência. Se num_chars for maior que o número de caracteres em uma sequência de texto, a sequência de texto inteira será retornada.

Exemplos

TEMP = STRING_TAIL(3, "JAN 15, 1997") Cria uma nova coluna denominada TEMP, que contém a sequência de texto ASCII "997".
TEMP = STRING_TAIL(10, "Pressure") Cria uma nova coluna denominada TEMP, que contém a sequência de texto ASCII "Pressure".
TEMP = STRING_TAIL(5, V1) Cria uma nova coluna denominada TEMP contendo os cinco últimos caracteres de cada sequência na coluna V1.
TEMP = STRING_TAIL(1, V1:V3) Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os últimos caracteres das sequências nas linhas correspondentes da coluna V1, os valores da coluna VX são os últimos caracteres das sequências nas linhas correspondentes da coluna V2, e os valores da coluna VY são os últimos caracteres das sequências nas linhas correspondentes de coluna V3.
TEMP = STRING_TAIL(12, V4[1:50]:V6] Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são os 12 últimos caracteres das sequências nas linhas 1-50 da coluna V1, os valores da coluna VX são os 12 últimos caracteres das sequências nas linhas 1-50 da coluna V2, e os valores da coluna VY são os 12 últimos caracteres das sequências nas linhas 1-50 da coluna V3.

Funções relacionadas

Função	Descrição
STRING_CONCAT	Concatena duas sequências de texto a partir dos intervalos de dados especificados
STRING_HEAD	Retorna os primeiros n caracteres de cada sequência no intervalo de dados especificado
STRING_LENGTH	Retorna o comprimento de cada sequência no intervalo de dados especificado.
STRING_SEG	Retorna o segmento da sequência entre dois índices especificados

Macro SUBSTR ou SUBSTRING

A está disponível no IBM Campaign e no IBM Interact. SUBSTR ou SUBSTRING

Sintaxe

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

Parâmetros

`string_value`

A sequência a partir da qual uma subsequência será executada.

`start_pos`

O caractere inicial a partir de cada subsequência será extraído.

nchars

O número de caracteres a serem extraídos (deve ser maior ou igual a 0). Se esse valor não for fornecido, todos os caracteres restantes em `string_value` serão extraídos.

Descrição

`SUBSTR` ou `SUBSTRING` extrai `nchars` caracteres da sequência, iniciando em `start_pos`. Se `nchars` for omitido, `SUBSTR` e `SUBSTRING` extrairá caracteres de `start_pos` até o fim da sequência. Espaços à direita são truncados automaticamente. Para evitar erros de sintaxe, assegure-se de separar os valores numéricos com uma vírgula e um espaço, conforme mostrado nos exemplos.

Importante: O IBM Interact suporta somente os formatos a seguir: `SUBSTR(string_value, start_pos[, nchars])` ou `SUBSTRING(string_value, start_pos[, nchars])`

Exemplos

SUBSTR SUBSTR Returns	("abcdef" FROM 1 FOR 2) ("abcdef", 1, 2) 'ab'
SUBSTR SUBSTR Returns	("abcdef" FROM -2 FOR 4) ("abcdef", -2, 4) 'a'
SUBSTR SUBSTR Returns	("abcdef" FROM 3) ("abcdef", 3) 'cdef'

Macro SUM

O `SUM` está disponível no IBM Campaign e no IBM Interact.

Sintaxe

`SUM(data [, keyword])`

Parâmetros

`data`

Os valores numéricos dos quais calcular a soma. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`keyword`

Essa palavra-chave opcional determina como o cálculo será executado sobre o intervalo de dados de entrada. Selecione uma das opções a seguir:

ALL - Executa o cálculo em todas as células em data (padrão)

COL - Executa o cálculo separadamente para cada coluna de data

ROW - Executa o cálculo separadamente para cada linha de data

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte “Especificações de Formato” na página 9.

Nota: Muitas funções de macro utilizam os parâmetros de palavra-chave {ALL | COL | ROW}. Estas palavras-chave não se aplicam ao IBM Campaign porque os dados de entrada são sempre uma coluna ou um campo único. A macro sempre se comportará como se a palavra-chave COL fosse especificada. Portanto, não é necessário especificar estas palavras-chave ao usar o IBM Campaign.

Descrição

SUM calcula a soma de todas as células no intervalo de dados especificado. Ele retorna uma única coluna.

Nota: SUM é o mesmo que a função macro TOTAL.

Exemplos

TEMP = SUM(3)Cria uma nova coluna denominada TEMP contendo o valor três.
TEMP = SUM((COLUMN(3, 5, 1)) Cria uma nova coluna denominada TEMP contendo o valor nove.
TEMP = SUM(V1) Cria uma nova coluna denominada TEMP contendo um valor único que é a soma do conteúdo da coluna V1.
TEMP = SUM(V1:V3) Cria uma nova coluna denominada TEMP contendo um valor único que é a soma do conteúdo das colunas V1, V2 e V3.
TEMP = SUM(V1[1:5]:V4) Cria uma nova coluna denominada TEMP contendo um valor único que é a soma das células nas linhas 10-20 das colunas V1 a V4.
TEMP = SUM(V1:V3, COL) Cria três novas colunas denominadas TEMP, VX e VY. O valor único na coluna TEMP é a soma do conteúdo da coluna V1, o valor único na coluna VX é a soma do conteúdo da coluna V2, e o valor único na coluna VY é a soma do conteúdo da coluna V3.
TEMP = SUM(V1[1:5]:V3, COL) Cria três novas colunas denominadas TEMP, VX e VY, cada uma contendo um valor único. O valor na coluna TEMP é a soma das células nas linhas 1-5 da coluna V1, o valor na coluna VX é a soma das células nas linhas 1-5 da coluna V2, e o valor na coluna VY é a soma das células nas linhas 1-5 da coluna V3.
TEMP = SUM(V1:V3, ROW) Cria uma nova coluna denominada TEMP, em que cada entrada na célula é a soma da linha correspondente nas colunas V1, V2 e V3.

`TEMP = SUM(V1[1:5]:V3, ROW)` Cria uma nova coluna denominada `TEMP`, em que as células nas linhas 1-5 contêm a soma da linha correspondente nas colunas `V1` a `V3`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`AVG` ou `MEAN`	Calcula a média aritmética ou a média de um intervalo de células

Macro TAN

O `TAN` está disponível apenas no IBM Campaign.

Sintaxe

```
TAN(data [, units_keyword])
```

Parâmetros

`data`

Os valores numéricos dos quais calcular a tangente. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`units_keyword`

Essa palavra-chave opcional determina se os valores de entrada e os resultados são interpretados como graus ou radianos. Selecione uma das opções a seguir:

`RADIAN` - Executa os cálculos em radianos (padrão)

`DEGREE` - Executa o cálculo em graus

Se esse parâmetro não for especificado, o padrão será radianos. (Para converter de radianos em graus, divida pelo `PI` e multiplique por 180).

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte “Especificações de Formato” na página 9.

Descrição

`TAN` calcula a tangente dos valores no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a tangente dos números na coluna de entrada correspondente.

Exemplos

`TEMP = TAN(PI/4)` ou `TEMP = TAN(PI/4, 0)` ou `TEMP = TAN(PI/4, RADIAN)` Cria uma nova coluna denominada `TEMP` contendo o valor um.

`TEMP = TAN(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é a tangente (em radianos) do conteúdo da coluna `V1`.
`TEMP = TAN(V1:V3, 1)` ou `TEMP = TAN(V1:V3, DEGREE)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são as tangentes do conteúdo da coluna `V1`, os valores da coluna `VX` são as tangentes do conteúdo da coluna `V2`, e os valores da coluna `VY` são as tangentes do conteúdo da coluna `V3`. Todos os valores estão em graus.
`TEMP = TAN(V1[1:5]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna `TEMP` são as tangentes das linhas correspondentes da coluna `V1`, e os valores na coluna `VX` são as tangentes das linhas correspondentes da coluna `V2`. Todos os valores estão em radianos.

Funções relacionadas

Função	Descrição
`ATAN`	Calcula o arco tangente do conteúdo do intervalo de dados especificado
`COS`	Calcula o cosseno do conteúdo do intervalo de dados especificado
`COT`	Calcula a cotangente do conteúdo do intervalo de dados especificado
`SIN`	Calcula o seno do conteúdo do intervalo de dados especificado
`TANH`	Calcula a tangente hiperbólica do conteúdo do intervalo de dados especificado

Macro TANH

A macro TANH está disponível apenas no IBM Campaign.

Sintaxe

```
TANH(data [, units_keyword])
```

Parâmetros

`data`

Os valores numéricos dos quais calcular a tangente hiperbólica. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`units_keyword`

Essa palavra-chave opcional determina se os valores de entrada e os resultados são interpretados como graus ou radianos. Selecione uma das opções a seguir:

RADIAN - Executa os cálculos em radianos (padrão)

DEGREE - Executa o cálculo em graus

Se esse parâmetro não for especificado, o padrão será radianos. (Para converter de radianos em graus, divida pelo PI e multiplique por 180).

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte "Especificações de Formato" na página 9.

Descrição

TANH calcula a tangente hiperbólica dos valores no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a tangente hiperbólica dos números na coluna de entrada correspondente. A tangente hiperbólica de um número é calculada conforme a seguir:

$$\tanh(x) = \frac{\sinh(x)}{\cosh(x)}$$

Nota: Se o valor de *x* for muito grande, um erro de estouro será retornado. Isso ocorrerá se o tanh (*x*) exceder o valor de vírgula flutuante máximo de 32 bits. Se cosh (*x*) for zero, o TANH retornará o valor de vírgula flutuante máximo de 32 bits.

Exemplos

`TEMP = TANH(PI)` ou `TEMP = TANH(PI, 0)` ou `TEMP = TANH(PI, RADIAN)` Cria uma nova coluna denominada TEMP contendo o valor um.
`TEMP = TANH(V1)` Cria uma nova coluna denominada TEMP, em que cada valor é a tangente hiperbólica (em radianos) do conteúdo da coluna V1.
`TEMP = TANH(V1:V3, 1)` ou `TEMP = TANH(V1:V3, DEGREE)` Cria três novas colunas denominadas TEMP, VX e VY. Os valores na coluna TEMP são as tangentes hiperbólicas do conteúdo da coluna V1, os valores da coluna VX são as tangentes hiperbólicas do conteúdo da coluna V2, e os valores da coluna VY são as tangentes hiperbólicas do conteúdo da coluna V3. Todos os valores estão em graus.
`TEMP = TANH(V1[1:5]:V2)` Cria duas novas colunas denominadas TEMP e VX, cada uma com valores nas linhas 1-5 (as outras células estão vazias). Os valores na coluna TEMP são as tangentes hiperbólicas das linhas correspondentes da coluna V1, e os valores na coluna VX são as tangentes hiperbólicas das linhas correspondentes da coluna V2. Todos os valores estão em radianos.

Funções relacionadas

Função	Descrição
ATAN	Calcula o arco tangente do conteúdo do intervalo de dados especificado
COSH	Calcula o cosseno hiperbólico do conteúdo do intervalo de dados especificado

Função	Descrição
COT	Calcula a cotangente do conteúdo do intervalo de dados especificado
SINH	Calcula o seno hiperbólico do conteúdo do intervalo de dados especificado
TAN	Calcula a tangente do conteúdo do intervalo de dados especificado

Macro TOTAL

O TOTAL está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
TOTAL(data [, keyword])
```

Parâmetros

data

Os valores numéricos dos quais calcular a soma. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

keyword

Essa palavra-chave opcional determina como o cálculo será executado sobre o intervalo de dados de entrada. Selecione uma das opções a seguir:

ALL - Executa o cálculo em todas as células em data (padrão)

COL - Executa o cálculo separadamente para cada coluna de data

ROW - Executa o cálculo separadamente para cada linha de data

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte “Especificações de Formato” na página 9.

Nota: Muitas funções de macro utilizam os parâmetros de palavra-chave {ALL | COL | ROW}. Estas palavras-chave não se aplicam ao IBM Campaign porque os dados de entrada são sempre uma coluna ou um campo único. A macro sempre se comportará como se a palavra-chave COL fosse especificada. Portanto, não é necessário especificar estas palavras-chave ao usar o IBM Campaign.

Descrição

TOTAL calcula a soma de todas as células no intervalo de dados especificado.

Nota: TOTAL é o mesmo que a função macro SUM.

Exemplos

`TEMP = TOTAL(3)`Cria uma nova coluna denominada `TEMP` contendo o valor três.
`TEMP = TOTAL((COLUMN(3, 5, 1))` Cria uma nova coluna denominada `TEMP` contendo o valor nove.
`TEMP = TOTAL(V1)` Cria uma nova coluna denominada `TEMP` contendo um valor único que é a soma do conteúdo da coluna `V1`.
`TEMP = TOTAL(V1:V3)` Cria uma nova coluna denominada `TEMP` contendo um valor único que é a soma do conteúdo das colunas `V1`, `V2` e `V3`.
`TEMP = TOTAL(V1[1:5]:V4)` Cria uma nova coluna denominada `TEMP` contendo um valor único que é a soma das células nas linhas 10-20 das colunas `V1` a `V4`.
`TEMP = TOTAL(V1:V3, COL)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. O valor único na coluna `TEMP` é a soma do conteúdo da coluna `V1`, o valor único na coluna `VX` é a soma do conteúdo da coluna `V2`, e o valor único na coluna `VY` é a soma do conteúdo da coluna `V3`.
`TEMP = TOTAL(V1[1:5]:V3, COL)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`, cada uma contendo um valor único. O valor na coluna `TEMP` é a soma das células nas linhas 1-5 da coluna `V1`, o valor na coluna `VX` é a soma das células nas linhas 1-5 da coluna `V2`, e o valor na coluna `VY` é a soma das células nas linhas 1-5 da coluna `V3`.
`TEMP = TOTAL(V1:V3, ROW)` Cria uma nova coluna denominada `TEMP`, em que cada entrada na célula é a soma da linha correspondente nas colunas `V1`, `V2` e `V3`.
`TEMP = TOTAL(V1[1:5]:V3, ROW)` Cria uma nova coluna denominada `TEMP`, em que as células nas linhas 1-5 contêm a soma da linha correspondente nas colunas `V1` a `V3`. Outras células em `TEMP` estão vazias.

Funções relacionadas

Função	Descrição
`AVG` ou `MEAN`	Calcula a média aritmética ou a média de um intervalo de células

Macro TRUNCATE

O `TRUNCATE` está disponível apenas no IBM Campaign.

Sintaxe

```
TRUNCATE(data)
```

Parâmetros

`data`

Os valores numéricos para truncar. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

`TRUNCATE` calcula a parte inteira de cada valor no intervalo de dados especificado. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a parte inteira (não fracionária) dos números na coluna de entrada correspondente.

Nota: As funções macro `FRACTION` e `TRUNCATE` são complementares na medida em que elas somam para os valores originais.

Exemplos

`TEMP = TRUNCATE(4.3)` Cria uma nova coluna denominada `TEMP` contendo o valor `4`.
`TEMP = TRUNCATE(2.9)` Cria uma nova coluna denominada `TEMP` contendo o valor `-2`.
`TEMP = TRUNCATE(V1)` Cria uma nova coluna denominada `TEMP`, em que cada valor é a parte fracionária do conteúdo da coluna `V1`.
`TEMP = TRUNCATE(V1:V3)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. Os valores na coluna `TEMP` são as partes truncadas da coluna `V1`, os valores da coluna `VX` são as partes truncadas da coluna `V2`, e os valores da coluna `VY` são as partes truncadas da coluna `V3`.
`TEMP = TRUNCATE(V1[10:20])` Cria uma nova coluna denominada `TEMP`, em que as 11 primeiras células contêm as partes truncadas dos valores nas linhas 10-20 da coluna `V1`. Outras células em `TEMP` estão vazias.
`TEMP = TRUNCATE(V1[50:99]:V2)` Cria duas novas colunas denominadas `TEMP` e `VX`, cada uma com valores nas linhas 1-50 (as outras células estão vazias). Os valores na coluna `TEMP` são as partes truncadas das linhas da coluna `V1`, e os valores na coluna `VX` são as partes truncadas dos valores na coluna `V2`.

Funções relacionadas

Função	Descrição
`CEILING`	Calcula o limite de cada valor no intervalo de dados especificado
`FLOOR`	Calcula o piso de cada valor no intervalo de dados especificado.
`FRACTION`	Retorna a parte fracionária de cada valor no intervalo de dados especificado.

Macro UPPER

O UPPER está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
UPPER(data)
```

Parâmetros

data

O valor de sequência a ser convertido em maiúsculas.

Descrição

UPPER converte cada valor de sequência no intervalo de dados especificado em maiúsculas. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a sequência em letras maiúsculas na coluna de entrada correspondente.

Exemplos

`Temp = UPPER "gold"` Cria uma nova coluna denominada Temp contendo "GOLD".
`TEMP = UPPER( "jan 15, 1997")` Cria uma nova coluna denominada TEMP, que contém a sequência de texto ASCII " JAN 15, 1997 ".
`TEMP = UPPER( "Pressure")` Cria uma nova coluna denominada TEMP, que contém a sequência de texto ASCII PRESSURE.
`TEMP = UPPER(V1)` Cria uma nova coluna denominada TEMP contendo caracteres maiúsculos de cada sequência na coluna V1.

Macro VARIANCE

O VARIANCE está disponível apenas no IBM Campaign.

Sintaxe

```
VARIANCE(data [, keyword])
```

Parâmetros

data

Os valores numéricos dos quais calcular a variância. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

keyword

Essa palavra-chave opcional determina como o cálculo será executado sobre o intervalo de dados de entrada. Selecione uma das opções a seguir:

`ALL` - Executa o cálculo em todas as células em `data` (padrão)

`COL` - Executa o cálculo separadamente para cada coluna de `data`

`ROW` - Executa o cálculo separadamente para cada linha de `data`

Para obter mais detalhes sobre como usar palavras-chave no IBM Campaign, consulte "Especificações de Formato" na página 9.

Nota: Muitas funções de macro utilizam os parâmetros de palavra-chave {`ALL` | `COL` | `ROW`}. Estas palavras-chave não se aplicam ao **IBM Campaign** porque os dados de entrada são sempre uma coluna ou um campo único. A macro sempre se comportará como se a palavra-chave `COL` fosse especificada. Portanto, não é necessário especificar estas palavras-chave quando usar **IBM Campaign** .

Descrição

`VARIANCE` calcula a variância de todos os valores no intervalo de dados especificado. A variância é o desvio padrão ao quadrado. A variância é calculada conforme a seguir:

$$\frac{1}{n-1}\sum_{j=1}^{n}(x_j - \text{mean})^2$$

em que *x* são as amostras, *n* é o número de amostras e *mean* é a média da distribuição.

Nota: Se o número de amostras for *n* = 1, `VARIANCE` retornará um erro.

Exemplos

`TEMP = VARIANCE(V1)` Cria uma nova coluna denominada `TEMP` contendo um valor único que é a variância do conteúdo da coluna `V1`.
`TEMP = VARIANCE(V1:V3)` Cria uma nova coluna denominada `TEMP`, em que cada entrada na célula é a variância da linha correspondente nas colunas `V1`, `V2`e `V3`.
`TEMP = VARIANCE(V1[10:20])` Cria uma nova coluna denominada `TEMP` contendo um valor único que é a variância das células nas linhas 10-20 da coluna `V1`.
`TEMP = VARIANCE(V1[1:5]:V4)` Cria uma nova coluna denominada `TEMP` contendo um valor único que é a variância das células nas linhas 1-5 das colunas `V1` a `V4`.

`TEMP = VARIANCE(V1:V3, COL)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`. O valor único na coluna `TEMP` é a variância do conteúdo da coluna `V1`, o valor único na coluna `VX` é a variância do conteúdo da coluna `V2`, e o valor único na coluna `VY` é a variância do conteúdo da coluna `V3`.
`TEMP = VARIANCE_(V1[1:5]:V3, COL)` ou `TEMP = VARIANCE(V1[1:5]:V3[1:5], COL)` Cria três novas colunas denominadas `TEMP`, `VX` e `VY`, cada uma contendo um valor único. O valor na coluna `TEMP` é a variância das células nas linhas 1-5 da coluna `V1`, o valor na coluna `VX` é a variância das células nas linhas 1-5 da coluna `V2`, e o valor na coluna `VY` é a variância das células nas linhas 1-5 da coluna `V3`.
`TEMP = VARIANCE(V1:V3, ROW)` Cria uma nova coluna denominada `TEMP`, em que cada entrada na célula é a variância da linha correspondente nas colunas `V1`, `V2`e `V3`.
`TEMP = VARIANCE(V1[1:5]:V3,ROW)` ou `TEMP = VARIANCE(V1[1:5]:V3[1:5], ROW)` Cria uma nova coluna denominada `TEMP`, em que as células nas linhas 1-5 contêm a variância da linha correspondente nas colunas `V1` a `V3`. Outras células em `TEMP` estão vazias.

Macro WEEKDAY

A macro WEEKDAY está disponível apenas no IBM Campaign.

Sintaxe

```
WEEKDAY(data [, conversion_keyword])
```

Parâmetros

`data`

As datas de texto ASCII para converter em valores numéricos representando dias da semana (1-7). Isso pode ser um texto ASCII entre aspas, uma coluna de texto, um intervalo de células contendo um texto ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de `data`, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

`conversion_keyword`

Esta palavra-chave opcional especifica como interpretar os formatos de texto para datas e horas. Selecione uma das opções a seguir:

`1 - mm/dd/yy (default)`

`2 - dd-mmm-yy`

`3 - mm/dd/yy hh:mm`

Se esse parâmetro não for especificado, o padrão será `1`.

Descrição

`WEEKDAY` converte valores de texto no intervalo de dados especificado em valores numéricos representando dias da semana usando o formato especificado para converter datas e horas. O número `0` para domingo, `1` para segunda-feira, e assim

por diante, até 6 para sábado. Se uma sequência de texto não puder ser analisada usando a conversion_keyword especificada, WEEKDAY retornará um erro.

Exemplos

TEMP = WEEKDAY("1/1/95") Cria uma nova coluna denominada TEMP contendo o número 0 (1º de janeiro de 1995 é um domingo).
TEMP = WEEKDAY(V1, 2) Cria uma nova coluna denominada TEMP contendo números para os dias da semana para as sequências de texto na coluna V1. Todas as sequências de texto na coluna V1 devem estar no formato dd-mmm-yy (caso contrário, ??? ' será retornado).
TEMP = WEEKDAY(V1:V3, 3) Cria três novas colunas denominadas TEMP, VX e VY. A coluna TEMP contém números que representam os dias da semana das sequências de texto na coluna V1. A coluna VX contém números que representam os dias da semana das sequências de texto na coluna V2. A coluna VY contém números que representam os dias da semana das sequências de texto na coluna V3. Todas as sequências de texto nas colunas V1 – V3 devem estar no formato mm/dd/yy hh:mm (caso contrário, ??? ' será retornado).
TEMP = WEEKDAY(V1[10:20]:V2, 10) Cria duas novas colunas denominadas TEMP e VX. A coluna TEMP contém os números que representam os dias da semana de sequências de texto nas linhas 10-20 da coluna V1. A coluna VX contém números que representam os dias da semana das sequências de texto nas linhas 10-20 na coluna V2. Todas as sequências de texto devem estar no formato mm/dd/yy (caso contrário, ??? ' será retornado).

Funções relacionadas

Função	Descrição
NUMBER	Converte sequências de texto ASCII de datas e horas em valores numéricos.

Macro WEEKDAYOF

A macro WEEKDAYOF está disponível apenas no IBM Campaign.

Sintaxe

```
WEEKDAYOF(date_string [, input_format])
```

Parâmetros

date_string

Um texto representando uma data válida.

input_format

Uma das palavras-chave na tabela abaixo, especificando o formato de data de date_string.

Descrição

WEEKDAYOF retorna o dia da semana como um número entre 0-6 (domingo 0, segunda-feira 1, e assim por diante) para a data especificada pelo date_string. Se input_format não for fornecido, a palavra-chave padrão DELIM_M_D_Y será utilizada.

Exemplos

WEEKDAYOF("08312000", MMDDAAAA) retorna o número 4, já que quinta-feira é o quarto dia da semana.

Nota: Consulte "Macro DATE" na página 51 para obter informações adicionais sobre formatos de data válidos.

Funções relacionadas

Função	Descrição
DAYOF	Retorna o dia do mês como um número.
MONTHOF	Retorna o mês do ano como um número.
YEAROF	Retorna o ano como um número.

Macro XOR

A macro XOR está disponível no IBM Campaign e no IBM Interact.

Sintaxe

```
data1 XOR data2
```

Parâmetros

data1

Os números inteiros não negativos para executar XOR bit a bit com os valores em data2. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

data2

Um ou mais números inteiros não negativos para executar XOR bit a bit com os valores em data1. Isso pode ser um valor constante, uma coluna, um intervalo de células ou uma expressão avaliada para qualquer item acima. O número de colunas em data2 deve ser igual ao número de colunas em data1, a menos que data2 seja uma constante. Para obter a definição de formato de data, consulte a seção "Parâmetros de Função de Macro" no capítulo nesse guia para seu produto IBM .

Descrição

XOR executa um XOR bit a bit entre dois intervalos de dados especificados. Ele retorna uma nova coluna para cada coluna de entrada, cada uma contendo a coluna correspondente em data1 com XOR bit a bit executado com a coluna

correspondente de data2 (ou seja, a primeira coluna de data1 tem o XOR bit a bit executado com a primeira coluna de data, a segunda coluna com a segunda coluna, e assim por diante).

Se data2 for uma constante, cada valor em data1 terá o XOR bit a bit executado por esse valor. Se data2 contiver uma ou mais colunas, os cálculos serão executados em uma base de linha por linha entre uma coluna data2 e uma coluna de data2. A primeira linha de data1 tem o XOR bit a bit executado com o valor da primeira linha de data2, a segunda linha com a segunda linha, e assim por diante. Esse cálculo de linha por linha produz um resultado para cada linha até o último valor da coluna mais curta.

Nota: A precisão para esta função macro é limitada a valores de número inteiro menores que 2^{24} . Nenhum valor negativo é permitido.

Exemplos

`TEMP = 3 XOR 7` Cria uma nova coluna denominada TEMP contendo o valor quatro (XOR bit a bit de 011 e 111 igual a 100).
`TEMP = V1 XOR 8` Cria uma nova coluna denominada TEMP, em que cada valor é o conteúdo da coluna V1, com o XOR bit a bit executado com o valor binário 1000.
`TEMP = V1 XOR V1` Cria uma nova coluna denominada TEMP contendo todos zeros (cada valor de XOR executado com ele mesmo produz zero).
`TEMP = V1 XOR V2` Cria uma nova coluna denominada TEMP, em que cada valor é o valor da linha da coluna V1 com XOR bit a bit executado com o valor da linha correspondente da coluna V2.
`TEMP = V1:V3 XOR V4:V6` Cria três novas colunas denominadas TEMP, VX e VY. A coluna TEMP contém os valores em V1 com o XOR bit a bit executado com os valores da linha correspondente da coluna V4. A coluna VX contém os valores de XOR bit a bit executado das colunas V2 e V5. A coluna VY contém os valores de XOR bit a bit executado das colunas V3 e V6.
`TEMP = V1[10:20] XOR V2` ou `TEMP = V1[10:20] XOR V2[1:11]` Cria uma nova coluna denominada TEMP, em que as 11 primeiras células contêm o resultado de XOR bit a bit executado dos valores nas linhas 10-20 da coluna V1 pelos valores nas linhas 1-11 da coluna V2. Outras células em TEMP estão vazias.

Funções relacionadas

Função	Descrição
BIT_AND	Calcula o AND bit a bit entre dois intervalos de dados especificados
BIT_NOT	Calcula o NOT bit a bit do conteúdo do intervalo de dados especificado
BIT_OR	Calcula o OR bit a bit entre dois intervalos de dados especificados

Macro YEAROF

A macro YEAROF está disponível apenas no IBM Campaign.

Sintaxe

```
YEAROF(date_string [, input_format])
```

Parâmetros

`date_string`

Um texto representando uma data válida.

`input_format`

Uma das palavras-chave na tabela abaixo, especificando o formato de data de `date_string`.

Descrição

`YEAROF` retorna o ano como um número para a data especificada pelo `date_string`. Se `input_format` não for fornecido, a palavra-chave padrão `DELIM_M_D_Y` será utilizada.

Exemplos

`YEAROF("31082000", DDMMAAAA)` retorna o número 2000.

Para obter informações adicionais sobre os formatos de data válidos, consulte "Macro DATE" na página 51.

Funções relacionadas

Função	Descrição
`DAYOF`	Retorna o dia do mês como um número.
`MONTHOF`	Retorna o mês do ano como um número.
`WEEKDAYOF`	Retorna o dia da semana como um número.

Entrando em contato com o suporte técnico do IBM

Se encontrar um problema que não possa ser resolvido consultando a documentação, o contato de suporte designado de sua empresa poderá registrar um chamado junto ao suporte técnico do IBM. Para assegurar que o seu problema seja resolvido com eficiência e êxito, colete informações antes de registrar seu chamado.

Se você não for um contato de suporte designado na sua empresa, entre em contato com o administrador do IBM para obter informações.

Informações a serem reunidas

Antes de entrar em contato com o suporte técnico do IBM, reúna as informações as seguir:

- Uma breve descrição da natureza de seu problema.
- Mensagens de erro detalhadas que você vê quando ocorre o problema.
- Etapas detalhadas para reproduzir o problema.
- Arquivos de log relacionados, arquivos de sessão, arquivos de configuração e arquivos de dados.
- Informações sobre seu produto e ambiente do sistema, que podem ser obtidas conforme descrito em "Informações do Sistema".

Informações do sistema

Ao ligar para o suporte técnico do IBM, você pode ser solicitado a fornecer informações sobre o seu ambiente.

Se o seu problema não impedi-lo de efetuar login, muitas dessas informações estarão disponíveis na página Sobre, que fornece informações sobre os aplicativos do IBM .

É possível acessar a página Sobre selecionando **Ajuda > Sobre**. Se a página Sobre não estiver acessível, é possível obter o número da versão de qualquer aplicativo do IBM ao visualizar o arquivo `version.txt` localizado no diretório de instalação para cada aplicativo.

Informações de contato para o suporte técnico do IBM

Para maneiras de entrar em contato com o suporte técnico do IBM. consulte o website de Suporte Técnico do Produto do IBM: (http://www.ibm.com/support/entry/portal/open_service_request).

Nota: Para inserir uma solicitação de suporte, deve-se efetuar login com uma conta do IBM. Se possível, essa conta deve estar vinculada ao seu número de cliente do IBM. Para saber mais sobre a associação da sua conta com seu número de cliente do IBM, consulte **Recursos de Suporte > Suporte de Software Autorizado** no Portal de Suporte.

© Copyright IBM Corp. 1998, 2015

Avisos

Estas informações foram desenvolvidas para produtos e serviços oferecidos nos Estados Unidos.

É possível que a IBM não ofereça os produtos, serviços ou recursos discutidos nesta publicação em outros países. Consulte um representante IBM local para obter informações sobre produtos e serviços disponíveis atualmente em sua área. Qualquer referência a produtos, programas ou serviços IBM não significa que apenas produtos, programas ou serviços IBM possam ser usados. Qualquer produto, programa ou serviço funcionalmente equivalente, que não infrinja nenhum direito de propriedade intelectual da IBM poderá ser usado em substituição a este produto, programa ou serviço da IBM. Entretanto, a avaliação e verificação da operação de qualquer produto, programa ou serviço não IBM são de responsabilidade do usuário.

A IBM pode ter patentes ou solicitações de patentes pendentes relativas a assuntos tratados nesta publicação. O fornecimento desta publicação não lhe garante direito algum sobre tais patentes. Pedidos de licença devem ser enviados, por escrito, para:

Gerência de Relações Comerciais e Industriais da IBM Brasil
Av. Pasteur, 138-146
Botafogo
Rio de Janeiro, RJ
CEP 22290-240

Para pedidos de licença relacionados a informações de DBCS (Conjunto de Caracteres de Byte Duplo), entre em contato com o Departamento de Propriedade Intelectual da IBM em seu país ou envie pedidos de licença, por escrito, para:

IBM World Trade Asia Corporation Licensing
2-31 Roppongi 3-chome,
Minato-ku
Tokyo 106,
Japan

O parágrafo a seguir não se aplica ao Reino Unido nem a nenhum país em que tais disposições não estejam de acordo com a legislação local: A INTERNATIONAL BUSINESS MACHINES CORPORATION FORNECE ESTA PUBLICAÇÃO "NO ESTADO EM QUE SE ENCONTRA", SEM GARANTIA DE NENHUM TIPO, SEJA EXPRESSA OU IMPLÍCITA, INCLUINDO, MAS A ELAS NÃO SE LIMITANDO, AS GARANTIAS IMPLÍCITAS OU CONDIÇÕES DE NÃO INFRAÇÃO, COMERCIALIZAÇÃO OU ADEQUAÇÃO A UM DETERMINADO PROPÓSITO. Alguns países não permitem a exclusão de garantias expressas ou implícitas em certas transações; portanto, essa disposição pode não se aplicar ao Cliente.

Essas informações podem conter imprecisões técnicas ou erros tipográficos. São feitas mudanças periódicas nas informações aqui contidas; tais mudanças serão incorporadas em futuras edições desta publicação. A IBM pode, a qualquer momento, aperfeiçoar e/ou alterar os produtos e/ou programas descritos nesta publicação, sem aviso prévio.

© Copyright IBM Corp. 1998, 2015

As referências nestas informações a websites não IBM são fornecidas apenas por conveniência e não representam de forma alguma um endosso a esses websites. Os materiais contidos nesses websites não fazem parte dos materiais desse produto IBM e o uso desses websites é de inteira responsabilidade do Cliente.

A IBM pode usar ou distribuir as informações fornecidas da forma que julgar apropriada sem incorrer em qualquer obrigação para com o Cliente.

Os licenciados deste programa que desejarem obter informações sobre este assunto com o propósito de permitir: (i) a troca de informações entre programas criados independentemente e outros programas (incluindo este) e (ii) o uso mútuo das informações trocadas, deverão entrar em contato com:

IBM Corporation
Gerência de Relações Comerciais e Industriais da IBM Brasil
Av. Pasteur, 138-146,
Botafogo, Rio de Janeiro, RJ
CEP 22290-240.

Tais informações podem estar disponíveis, sujeitas aos termos e condições apropriados, incluindo, em alguns casos, o pagamento de uma taxa.

O programa licenciado descrito nesta publicação e todo o material licenciado disponível são fornecidos pela IBM sob os termos do Contrato com o Cliente IBM, do Contrato Internacional de Licença do Programa IBM ou de qualquer outro contrato equivalente.

Todos os dados de desempenho aqui contidos foram determinados em um ambiente controlado. Portanto, os resultados obtidos em outros ambientes operacionais podem variar significativamente. Algumas medidas podem ter sido tomadas em sistemas em nível de desenvolvimento e não há garantia de que estas medidas serão iguais em sistemas geralmente disponíveis. Além disso, algumas medidas podem ter sido estimadas por extrapolação. Os resultados reais podem variar. Os usuários deste documento devem verificar os dados aplicáveis para seu ambiente específico.

As informações relativas a produtos não IBM foram obtidas junto aos fornecedores dos respectivos produtos, a partir de seus anúncios publicados ou de outras fontes disponíveis publicamente. A IBM não testou estes produtos e não pode confirmar a precisão de seu desempenho, compatibilidade nem qualquer outra reivindicação relacionada a produtos não IBM. As dúvidas sobre os recursos de produtos não IBM devem ser encaminhadas diretamente aos seus fornecedores.

Todas as declarações relacionadas aos objetivos e intenções futuras da IBM estão sujeitas a mudanças ou cancelamento sem aviso prévio e representam apenas metas e objetivos.

Todos os preços IBM mostrados são preços de varejo sugeridos pela IBM, são atuais e estão sujeitos a mudança sem aviso prévio. Os preços do revendedor podem variar.

Estas informações contêm exemplos de dados e relatórios usados nas operações diárias de negócios. Para ilustrá-los da forma mais completa possível, os exemplos podem incluir nomes de indivíduos, empresas, marcas e produtos. Todos estes nomes são fictícios e qualquer semelhança com nomes e endereços usados por uma empresa real é mera coincidência.

LICENÇA DE COPYRIGHT:

Estas informações contêm programas de aplicativos de amostra na linguagem fonte, ilustrando as técnicas de programação em diversas plataformas operacionais. O Cliente pode copiar, modificar e distribuir estes programas de amostra sem a necessidade de pagar à IBM, com objetivos de desenvolvimento, uso, marketing ou distribuição de programas de aplicativos em conformidade com a interface de programação de aplicativo para a plataforma operacional para a qual os programas de amostra são criados. Esses exemplos não foram testados completamente em todas as condições. Portanto, a IBM não pode garantir ou implicar a confiabilidade, manutenção ou função destes programas. Os programas de amostra são fornecidos "NO ESTADO EM QUE SE ENCONTRAM", sem garantia de nenhum tipo. A IBM não se responsabiliza por danos causados pelo uso dos programas de amostra.

Se estiver visualizando essas informações em formato eletrônico, as fotografias e ilustrações coloridas podem não aparecer.

Marcas Comerciais

IBM, o logotipo IBM e ibm.com are são marcas comerciais ou marcas registradas da International Business Machines Corp., registradas em vários países no mundo todo. Outros nomes de produtos e serviços podem ser marcas comerciais da IBM ou de outras empresas. Uma lista atual de marcas comerciais da IBM está disponível na Web em "Copyright and trademark information" em www.ibm.com/legal/copytrade.shtml.

Política de Privacidade e Termos de Considerações de Uso

Os produtos de Software IBM, incluindo software como soluções de serviço, ("Ofertas de Software") podem usar cookies ou outras tecnologias para coletar informações de uso do produto para ajudar a melhorar a experiência do usuário final, customizar interações com o usuário final ou para outros propósitos. Um cookie é uma parte de dados que um website pode enviar ao seu navegador, que, em seguida, pode ser armazenado em seu computador como uma tag que identifica o computador. Em muitos casos, nenhuma informação pessoal é coletada por esses cookies. Se uma Oferta de Software, que esteja sendo usada, permitir que você colete informações pessoais através de cookies e tecnologias semelhantes, nós o informaremos sobre as especificações abaixo.

Dependendo das configurações implementadas, esta Oferta de Software poderá usar cookies persistentes e de sessão que coletam o nome de cada usuário, e outras informações pessoais para propósitos de gerenciamento de sessão, usabilidade de usuário aprimorada ou outros propósitos funcionais ou de controle de uso. Esses cookies podem ser desativados, mas a desativação também eliminará a funcionalidade que eles ativam.

Várias jurisdições regulam a coleta de informações pessoais por meio de cookies e tecnologias semelhantes. Se as configurações implementadas para esta Oferta de Software fornecerem a você, como cliente, a capacidade de coletar informações pessoais de usuários finais por meio de cookies e outras tecnologias, é necessário procurar o seu próprio conselho jurídico sobre todas as leis aplicáveis a essa coleta de dados, incluindo quaisquer requisitos para o fornecimento de aviso e consentimento quando apropriado.

A IBM requer que os Clientes (1) forneçam um link claro e evidente para os termos de uso do website do Cliente (por exemplo, política de privacidade) que inclui um

link para a coleção de dados da IBM e do Cliente e práticas de uso, (2) notifiquem que cookies e gifs/web beacons claros estão sendo colocados no computador do visitante pela IBM em nome do Cliente juntamente com uma explicação do propósito de tal tecnologia, e (3) até a extensão requerida por lei, obtenham o consentimento dos visitantes do website antes de se colocar cookies e gifs/web beacons claros pelo Client ou IBM em nome do Cliente nos dispositivos do visitante do website

Para obter mais informações sobre o uso de várias tecnologias, incluindo cookies, para estes fins, consulte Declaração de Privacidade Online da IBM em http://www.ibm.com/privacy/details/us/en na seção autorizada "Cookies, Web Beacons e Outras Tecnologias."

Impresso no Brasil