

IBM Makros für IBM EMM
Version 9 Release 0
11. Dezember 2012

Benutzerhandbuch

IBM

Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen unter „Bemerkungen“ auf Seite 257 gelesen werden.

Diese Edition bezieht sich auf Version 9, Release 0, Modifikation 0 der IBM Kampagne-Produktfamilie und alle nachfolgenden Releases und Modifikationen, bis dieser Hinweis in einer neuen Edition geändert wird.

© Copyright IBM Corporation 1998, 2012.

Inhaltsverzeichnis

Kapitel 1. Verwendung von Makros in IBM Kampagne. 1

Zusammenfassungen der Makrofunktionen	1
Statistische Funktionen	1
Mathematische und trigonometrische Funktionen	2
Zeichenfolgefunktionen	6
Datums- und Zeitfunktionen	7
Gruppierungsfunktionen	8
Verschiedene Funktionen	8
Makrofunktionsparameter	9
Formatspezifikationen	9
Verwendung von Konstanten	10

Kapitel 2. Verwendung von Makros in IBM Interact 11

Zusammenfassungen der Makrofunktionen	11
Statistische Funktionen	11
Mathematische und trigonometrische Funktionen	12
Zeichenfolgefunktionen	13
Datums- und Zeitfunktionen	14
Verschiedene Funktionen	14
Makrofunktionsparameter	15
Formatspezifikationen	15
Verwendung von Konstanten	15

Kapitel 3. Verwendung von Makros in IBM PredictiveInsight 17

Zusammenfassungen der Makrofunktionen	17
Statistische Funktionen	17
Mathematische und trigonometrische Funktionen	19
Entwicklungsfunktionen	22
Zeichenfolgefunktionen	23
Datums- und Zeitfunktionen	24
Gruppierungsfunktionen	25
Verschiedene Funktionen	25
Hinweise zu Makroreferenzseiten	26
Makrofunktionsparameter	27
Formatspezifikationen	27
Verwendung von Zellen- und Spaltenbereichen	27
Verwendung von Konstanten	28
Rechengenauigkeit	29
Ergebnisse ungültiger Zellen und leere Zellen	30

Kapitel 4. IBM -Makroreferenz 33

ABS	33
ACOS	34
ACOT	36
ADD_MONTHS	37
ALIGN	38
AND	39
ASIN	40
ATAN	42
AVG	44
AVG_DEV	46

BETWEEN	48
BIT_AND	48
BIT_NOT	50
BIT_OR	51
BIT_XOR	53
BUFFER	55
CEILING	56
COLUMN	57
CONSTANT	59
COS	60
COSH	62
COT	64
COUNT	65
COUNT_DIFF	66
COV	68
CURRENT_DATE	69
CURRENT_DAY	70
CURRENT_JULIAN	70
CURRENT_MONTH	71
CURRENT_TIME	72
Datumseinstellung in Ihrer Webanwendung	72
CURRENT_WEEKDAY	73
CURRENT_YEAR	74
CV_FOLDS	74
DATALINK	76
DATE	77
DATE_FORMAT	80
DATE_JULIAN	81
DATE_STRING	82
DAY_BETWEEN	83
DAY_FROMNOW	84
DAY_INTERVAL	85
DAYOF	86
DDELINK	86
DECIMATE	88
DELAY	89
DERIVATIVE	90
DISTANCE	91
DISTINCT	92
DIV	93
EQ	95
EXP	96
EXTERNALCALLOUT	98
EXTRACT	98
FACTORIAL	100
FLOOR	101
FORMAT	102
FRACTION	105
GAUSS	106
GAUSS_AREA	108
GE	111
GRID	112
GROUPBY	114
GROUPBY_WHERE	117
GT	118
HISTOGRAM	119

IF	121	ROTATE_LEFT	203
IN	123	ROTATE_RIGHT	204
INIT	124	ROUND	205
INT	125	ROWNUM	206
INTEGRAL	126	RTRIM	206
INVERSE	128	SAMPLE_RANDOM	207
IS	129	SELECT	208
ISERROR	130	SIGN	210
ISEVEN	131	SIN	211
ISMEMBER	132	SINH	213
ISODD	133	SKEW	214
KURTOSIS	135	SLIDE_WINDOW	216
LAG	137	SORT	218
LE	138	SQRT	220
LIKE	140	STAT	221
LN oder LOG	142	STDV oder STDEV	223
LOG2	143	STRING_CONCAT	225
LOG10	144	STRING_HEAD	227
LOWER	145	STRING_LENGTH	228
LT	146	STRING_PROPER	229
LTRIM	148	STRING_SEG	230
MAX	148	STRING_TAIL	231
MAXINDEX	150	SUBSAMPLE	232
MEAN	152	SUBSTITUTE	234
MERGE	154	SUBSTR oder SUBSTRING	235
MIN	155	SUM	236
MINUS	157	TAN	238
MOD	158	TANH	239
MONTHOF	160	TO	241
MULT	161	TOTAL	242
NE	163	TRANPOSE	243
NORM_MINMAX	164	TRUNCATE	245
NORM_SIGMOID	168	UPPER	246
NORM_ZSCORE	171	VARIANCE	246
NOT	175	WEEKDAY	249
NPV	176	WEEKDAYOF	250
NUMBER	178	XOR	251
OFFSET	186	XTAB	252
OR	187	YEAROF	254
PCA	189		
PCA_FEATURES	190	Kontakt zum technischen Support von	
POSITION	192	IBM	255
PLUS	194		
POW	195	Bemerkungen	257
RANDOM	197	Marken	259
RANDOM_GAUSS	198	Hinweise zu Datenschutzrichtlinien und Nutzungs-	
RANK	200	bedingungen	259
REPEAT	201		

Kapitel 1. Verwendung von Makros in IBM Kampagne

Dieses Kapitel enthält Informationen zur Syntax der Makros, die in IBM® Kampagne zur Verfügung stehen. Alle Benutzer von IBM Kampagne sollten dieses Kapitel lesen, bevor sie die übrigen Kapitel des vorliegenden Handbuchs verwenden.

Zusammenfassungen der Makrofunktionen

Die Tabellen in diesem Abschnitt fassen die Makrofunktionen nach Kategorien zusammen.

Wichtig: Nicht alle Makros, die in diesem Handbuch aufgeführt sind, sind in IBM Kampagne verfügbar. Makros, die nur in IBM PredictiveInsight verfügbar sind, sind durch folgendes Symbol gekennzeichnet:  Makros, die nur in IBM Interact verfügbar sind, sind durch folgendes Symbol gekennzeichnet: 

Ausführliche Referenzseiten zu den einzelnen Makrofunktionen können in alphabetischer Reihenfolge von Kapitel 4, „IBM -Makroreferenz“, auf Seite 33 aufgerufen werden. „Makrofunktionsparameter“ auf Seite 27 enthält Informationen zu den Eingabeparametern der Makrofunktionen.

Zugehörige Informationen:

„Statistische Funktionen“

„Mathematische und trigonometrische Funktionen“ auf Seite 2

„Zeichenfolgefunktionen“ auf Seite 6

„Datums- und Zeitfunktionen“ auf Seite 7

„Gruppierungsfunktionen“ auf Seite 8

„Verschiedene Funktionen“ auf Seite 8

Statistische Funktionen

Makroname	Rückgabe	Beschreibung
AVG	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das arithmetische Mittel oder den Durchschnitt eines Zellenbereichs.
COUNT	Einzelwert in einer neuen Spalte	Zählt die Anzahl der Werte in einem angegebenen Datenbereich
MAX	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das Maximum eines Zellenbereichs.

Makroname	Rückgabe	Beschreibung
MEAN	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das arithmetische Mittel oder den Durchschnitt eines Zellenbereichs.
MIN	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das Minimum eines Zellenbereichs.
STDV oder STDEV	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Standardabweichung eines Zellenbereichs.
VARIANCE	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Varianz eines Zellenbereichs.

Mathematische und trigonometrische Funktionen

Makroname	Rückgabe	Beschreibung
ABS	Eine Spalte für jede Eingabespalte	Berechnet den absoluten Wert des Inhalts des angegebenen Datenbereichs.
ACOS	Eine Spalte für jede Eingabespalte	Berechnet den Arkuskosinus des Inhalts des angegebenen Datenbereichs.
ACOT	Eine Spalte für jede Eingabespalte	Berechnet den Arkuskotangens des Inhalts des angegebenen Datenbereichs.
ASIN	Eine Spalte für jede Eingabespalte	Berechnet den Arkussinus des Inhalts des angegebenen Datenbereichs.
ATAN	Eine Spalte für jede Eingabespalte	Berechnet den Arkustangens des Inhalts des angegebenen Datenbereichs.

Makroname	Rückgabe	Beschreibung
AVG	Eine Spalte für jede Eingabespalte	Berechnet das arithmetische Mittel oder den Durchschnitt der Zellen im angegebenen Datenbereich.
BETWEEN	Eine Spalte für jede Eingabespalte	Vergleicht zwei Werte, um festzustellen, ob der angegebene Wert zwischen zwei anderen Werten liegt.
CEILING	Eine Spalte für jede Eingabespalte	Berechnet die Obergrenze jedes Werts im angegebenen Datenbereich.
COLUMN	Eine Spalte für jede Eingabespalte	Erstellt neue Spalten, wobei die Eingabewerte in jeder Spalte vertikal verkettet werden.
COS	Eine Spalte für jede Eingabespalte	Berechnet den Kosinus des Inhalts des angegebenen Datenbereichs.
COSH	Eine Spalte für jede Eingabespalte	Berechnet den Hyperbelkosinus des Inhalts des angegebenen Datenbereichs.
COT	Eine Spalte für jede Eingabespalte	Berechnet den Kotangens des Inhalts des angegebenen Datenbereichs.
COUNT	Eine Spalte mit einem Einzelwert	Zählt die Anzahl der Zellen, die Werte enthalten, im angegebenen Datenbereich.
EXP	Eine Spalte für jede Eingabespalte	Erhebt die natürliche Zahl (e) in die Potenz, die durch den Inhalt jeder Zelle im angegebenen Datenbereich angegeben ist.
FACTORIAL	Eine Spalte für jede Eingabespalte	Berechnet die Fakultät jedes Werts im angegebenen Datenbereich.
FLOOR	Eine Spalte für jede Eingabespalte	Berechnet die Untergrenze jedes Werts im angegebenen Datenbereich.
FRACTION	Eine Spalte für jede Eingabespalte	Berechnet die Nachkommastellen jedes Werts im angegebenen Datenbereich.
INT	Eine Spalte für jede Eingabespalte	Berechnet den abgerundeten Ganzzahlwert des Inhalts des angegebenen Datenbereichs.
INVERSE	Eine Spalte für jede Eingabespalte	Berechnet den Negativwert des Inhalts des angegebenen Datenbereichs.
LN	Eine Spalte für jede Eingabespalte	Berechnet den natürlichen Logarithmus des Inhalts des angegebenen Datenbereichs.

Makroname	Rückgabe	Beschreibung
LOG	Eine Spalte für jede Eingabespalte	Berechnet den natürlichen Logarithmus des Inhalts des angegebenen Datenbereichs.
LOG2	Eine Spalte für jede Eingabespalte	Berechnet den binären Logarithmus des Inhalts des angegebenen Datenbereichs.
LOG10	Eine Spalte für jede Eingabespalte	Berechnet den dekadischen Logarithmus des Inhalts des angegebenen Datenbereichs.
MAX	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das Maximum eines Zellenbereichs.
MEAN	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das arithmetische Mittel oder den Durchschnitt eines Zellenbereichs.
MIN	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das Minimum eines Zellenbereichs.
RANDOM	Eine Spalte mit der angegebenen Anzahl von Werten	Gibt die angegebene Anzahl von Zufallszahlen zurück.
RANDOM_GAUSS	Eine Spalte mit der angegebenen Anzahl von Werten	Gibt die angegebene Anzahl von Zufallswerten aus einer gaußschen Verteilung zurück.
ROUND	Eine Spalte für jede Eingabespalte	Berechnet den gerundeten Wert des Inhalts des angegebenen Datenbereichs.
SIGN	Eine Spalte für jede Eingabespalte	Berechnet das Vorzeichen (positiv oder negativ) der Werte im angegebenen Datenbereich.
SIN	Eine Spalte für jede Eingabespalte	Berechnet den Sinus des Inhalts des angegebenen Datenbereichs.
SINH	Eine Spalte für jede Eingabespalte	Berechnet den Hyperbelsinus des Inhalts des angegebenen Datenbereichs.

Makroname	Rückgabe	Beschreibung
SQRT	Eine Spalte für jede Eingabespalte	Berechnet die Quadratwurzel des Inhalts des angegebenen Datenbereichs.
STDV oder STDEV	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Standardabweichung eines Zellenbereichs.
SUM	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Summe eines Zellenbereichs.
TAN	Eine Spalte für jede Eingabespalte	Berechnet den Tangens des Inhalts des angegebenen Datenbereichs.
TANH	Eine Spalte für jede Eingabespalte	Berechnet den Hyperbeltangens des Inhalts des angegebenen Datenbereichs.
TOTAL	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Summe eines Zellenbereichs.
TRUNCATE	Eine Spalte für jede Eingabespalte	Berechnet den Ganzzahlanteil jedes Werts im angegebenen Datenbereich.
VARIANCE	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Varianz eines Zellenbereichs.

Zeichenfolgefunktionen

Makroname	Rückgabe	Beschreibung
FORMAT	Eine Spalte für jede Eingabespalte	Steuert die Ausgabeformatierung für Zahlen und Zeichenfolgen (z. B. Ausgabebreite, Ausrichtung, numerische Genauigkeit, Dezimalzeichen, Trennzeichen usw.). Gibt die formatierte Ausgabezeichenfolge zurück.
LIKE	Eine Spalte für jede Eingabespalte	Ermittelt, ob eine Zeichenfolge mit einem angegebenen Muster übereinstimmt.
LOWER	Eine Spalte für jede Eingabespalte	Konvertiert Zeichenfolgewerte in Kleinschreibung.
LTRIM	Eine Spalte für jede Eingabespalte	Entfernt führende Leerzeichen aus jedem Zeichenfolgewart.
NUMBER	Eine Spalte für jede Eingabespalte	Konvertiert ASCII-Zeichenfolgen für Uhrzeit- und Datumsangaben in numerische Werte.
POSITION	Eine Spalte für jede Eingabespalte	Gibt die Anfangsposition eines Musters in einer Zeichenfolge zurück.
RTRIM	Eine Spalte für jede Eingabespalte	Entfernt nachfolgende Leerzeichen aus jedem Zeichenfolgewart.
STRING_CONCAT	Eine Spalte mit einem Wert für jede Zeile der kürzesten Eingabespalte	Verkettet Zeichenfolgen aus den angegebenen Datenbereichen.
STRING_HEAD	Eine Spalte für jede Eingabespalte	Gibt die ersten n Zeichen jeder Zeichenfolge im angegebenen Datenbereich zurück.
STRING_LENGTH	Eine Spalte für jede Eingabespalte	Gibt die Länge jeder Zeichenfolge im angegebenen Datenbereich zurück.
STRING_PROPER	Eine Spalte für jede Eingabespalte	Konvertiert jeden Zeichenfolgewart so, dass der erste Buchstabe oder irgendein Buchstabe, der auf ein Leerzeichen oder Symbol (außer dem Unterstrich) folgt, in einen Großbuchstaben und alle anderen Zeichen in Kleinbuchstaben geändert werden.
STRING_SEG	Eine Spalte für jede Eingabespalte	Gibt das Zeichenfolgesegment zwischen zwei angegebenen Indizes zurück.

Makroname	Rückgabe	Beschreibung
STRING_TAIL	Eine Spalte für jede Eingabespalte	Gibt die letzten n Zeichen jeder Zeichenfolge im angegebenen Datenbereich zurück.
SUBSTR oder SUBSTRING	Eine Spalte für jede Eingabespalte	Gibt Zeichen aus einer Zeichenfolge ab einer Anfangsposition zurück.
UPPER	Eine Spalte für jede Eingabespalte	Konvertiert Zeichenfolgewerte in Großschreibung.

Datums- und Zeitfunktionen

Makroname	Rückgabe	Beschreibung
ADD_MONTHS	Eine Spalte für jede Eingabespalte	Gibt nach Addition der angegebenen Anzahl von Monaten ein Datum zurück.
CURRENT_DATE	Eine Spalte für jede Eingabespalte	Gibt das aktuelle Datum in einem bestimmten Format zurück.
CURRENT_DAY	Eine Spalte für jede Eingabespalte	Gibt den aktuellen Tag des Monats als Zahl zwischen 1 und 31 zurück.
CURRENT_JULIAN	Eine Spalte für jede Eingabespalte	Gibt die julianische Zahl für das aktuelle Datum zurück.
CURRENT_MONTH	Eine Spalte für jede Eingabespalte	Gibt den aktuellen Monat des Jahres als Zahl zwischen 1 und 12 zurück.
CURRENT_TIME	Eine Spalte für jede Eingabespalte	Gibt die aktuelle Uhrzeit als Zeichenfolge zurück.
CURRENT_WEEKDAY	Eine Spalte für jede Eingabespalte	Gibt den aktuellen Wochentag als Zahl zwischen 0 und 6 zurück.
CURRENT_YEAR	Eine Spalte für jede Eingabespalte	Gibt das aktuelle Jahr als Zahl zurück.
DATE	Eine Spalte für jede Eingabespalte	Konvertiert eine Datumszeichenfolge in ein julianisches Datum.
DATE_FORMAT	Eine Spalte für jede Eingabespalte	Wandelt Datumsformate um.
DATE_JULIAN	Eine Spalte für jede Eingabespalte	Gibt das julianische Datum zurück.
DATE_STRING	Eine Spalte für jede Eingabespalte	Gibt die Datumszeichenfolge des julianischen Datums zurück.
DAY_BETWEEN	Eine Spalte für jede Eingabespalte	Gibt die Anzahl der Tage zwischen zwei Terminen zurück.

Makroname	Rückgabe	Beschreibung
DAY_FROMNOW	Eine Spalte für jede Eingabespalte	Gibt die Anzahl der Tage vom aktuellen Datum bis zum angegebenen Datum zurück.
DAY_INTERVAL	Eine Spalte für jede Eingabespalte	Gibt die Anzahl der Tage zwischen zwei Terminen zurück.
DAYOF	Eine Spalte für jede Eingabespalte	Gibt den Tag des Monats als Zahl zurück.
MONTHOF	Eine Spalte für jede Eingabespalte	Gibt den Monat des Jahres als Zahl zurück.
WEEKDAY	Eine Spalte für jede Eingabespalte	Konvertiert ASCII-Datumszeichenfolgen in Wochentage.
WEEKDAYOF	Eine Spalte für jede Eingabespalte	Gibt den Wochentag der Woche als Zahl zurück.
YEAROF	Eine Spalte für jede Eingabespalte	Gibt das Jahr als Zahl zurück.

Gruppierungsfunktionen

Makroname	Rückgabe	Beschreibung
GROUPBY	Eine neue Spalte mit einem Wert für jede Zeile	Fasst mehrere Zeilen von Daten in einer Gruppe zusammen.
GROUPBY_WHERE	Eine neue Spalte mit einem Wert für jede Zeile	Fasst mehrere Zeilen von Daten zusammen, die eine bestimmte Bedingung erfüllen und sich in einer Gruppe befinden.

Verschiedene Funktionen

Makroname	Rückgabe	Beschreibung
IF	Eine Spalte mit einem Wert für jede Zeile der kürzesten Eingabespalte	Leitet eine If-then-else-Anweisung einer bedingten Anweisung ein.
ISERROR	Eine Spalte mit einem Wert für jede Zeile der kürzesten Eingabespalte	Gibt eine Eins zurück, wenn ein Wert in der Eingabezeile eine fehlerhafte Zelle (???) enthält, andernfalls eine Null.
ISEVEN	Eine Spalte für jede Eingabespalte	Prüft, ob die Eingabewerte gerade (d. h. durch zwei teilbar) sind.
ISODD	Eine Spalte für jede Eingabespalte	Prüft, ob die Eingabewerte ungerade (d. h. nicht durch zwei teilbar) sind.

Makroname	Rückgabe	Beschreibung
ROWNUM	Eine Spalte für jede Eingabespalte	Generiert fortlaufende Zahlen von eins bis zur Anzahl der Einträge

Makrofunktionsparameter

Dieser Abschnitt beschreibt die Parameter und ihre Verwendung für Makrofunktionen in IBM Kampagne.

Formatspezifikationen

Dieser Abschnitt beschreibt das Format für einige häufig verwendete Parameter. Es gilt für alle Referenzen auf diese Parameter durch Makrofunktionsspezifikationen in diesem Kapitel.

Daten

Der Parameter Daten stellt eine Datenspalte dar, auf die eine Makrofunktion angewendet werden soll. Es kann sich um eine Konstante oder um ein Feld handeln. Nähere Einzelheiten finden Sie im Abschnitt zur jeweiligen Makrofunktion.

Anmerkung: IBM Kampagne unterstützt keine Berechnungen in mehreren Feldern gleichzeitig oder in einer Untergruppe von Zeilen, wie dies in IBM Predictive-Insight möglich ist.

Einige weitere Parameternamen verwenden dasselbe Format wie Daten. Die Beschreibungen dieser Parameter verweisen auf dieses Kapitel und Format.

Schlüsselwort

Der Parameter Schlüsselwort steuert das Verhalten der Makrofunktion. Er zeigt an, dass ein Schlüsselwort angegeben werden kann (wenn es weggelassen wird, wird der Standardwert verwendet). Die Schlüsselwortoptionen werden für die jeweilige Makrofunktion in der folgenden Form aufgeführt:

{**Option1** | Option2 | Option3}

Wählen Sie die Schlüsselwortoption aus, die zu dem gewünschten Verhalten führt. Die Standardoption wird in Fettschrift angezeigt. Es können zum Beispiel die folgenden Optionen angegeben sein:

{**RADIANS** | DEGREES}

In diesem Fall sind die beiden folgenden Makrofunktionen gültig:

COS(W1, RADIANS) und COS(W1, DEGREES)

Anmerkung: Viele Makrofunktionen nehmen die Schlüsselwortparameter {ALL | COL | ROW} an. Diese Schlüsselwörter gelten nicht für IBM Kampagne, da es sich bei den Eingabedaten immer um eine Einzelspalte oder ein Einzelfeld handelt. Das Makro verhält sich immer so, als ob das Schlüsselwort COL angegeben würde. Deshalb brauchen Sie diese Schlüsselwörter nicht anzugeben, wenn Sie IBM Kampagne verwenden.

Verwendung von Konstanten

Die meisten Makrofunktionsparameter verarbeiten numerische Konstanten oder Ausdrücke, deren Auswertung eine numerische Konstante ergibt (Makrofunktionen für Zeichenfolgen können Zeichenfolgekonstanten verarbeiten).

In Makrofunktionen, die alle Einträge einzeln durchgehen (etwa bei der Addition zweier numerischer Spalten), ist die Verwendung einer Konstante äquivalent zur Angabe einer Spalte, die in jeder Zeile den betreffenden konstanten Wert enthält. Im Wesentlichen wird eine Konstante, die als ein Eingabeparameter angegeben ist, bis zur Länge der Eingabe expandiert.

Einige Makrofunktionen können ASCII-Zeichenfolgen und numerische Konstanten verarbeiten. Bei Parametern, die sowohl numerische Konstanten als auch ASCII-Zeichenfolgen verarbeiten können, ist dies im Abschnitt "Parameter" der jeweiligen Makrofunktion vermerkt.

Die folgende Tabelle enthält einige Beispiele.

Funktionsdefinition	Interpretation der Konstante
<code>PERCENT_UTILIZ = (CURR_BAL*100)/ CREDIT_LIM</code>	Die Konstante 100 wird als Spalte interpretiert, die dieselbe Zeilenzahl wie die Spalte CURR_BAL enthält, wobei jede Zeile die Konstante 100 enthält. Das abgeleitete Feld PERCENT_UTILIZ enthält jeden Wert von CURR_BAL, multipliziert mit 100 und dividiert durch jeden Wert von CREDIT_LIM.
<code>NAME = STRING_CONCAT ("Herr ", NACHNAME)</code>	Die Konstante "Herr " wird als Spalte interpretiert, die dieselbe Zeilenzahl wie die Spalte NACHNAME enthält, wobei jede Zeile die Konstante "Herr " enthält. Im abgeleiteten Feld NAME ist jeder der Zeichenfolgen in NACHNAME die Zeichenfolge "Herr " vorangestellt.

Kapitel 2. Verwendung von Makros in IBM Interact

Dieser Abschnitt enthält Informationen zur Syntax der Makros, die in IBM Interact zur Verfügung stehen. Alle Benutzer von IBM Interact sollten diesen Abschnitt lesen, bevor sie die übrigen Abschnitte des vorliegenden Handbuchs verwenden.

Zusammenfassungen der Makrofunktionen

Die Tabellen in diesem Abschnitt fassen die Makrofunktionen nach den folgenden Kategorien zusammen:

Wichtig: Nicht alle Makros, die in diesem Handbuch aufgeführt sind, sind in IBM Interact verfügbar. Makros, die nur in IBM PredictiveInsight verfügbar sind, sind durch folgendes Symbol gekennzeichnet: 

Ausführliche Referenzseiten zu den einzelnen Makrofunktionen können in alphabetischer Reihenfolge von Kapitel 4, „IBM -Makroreferenz“, auf Seite 33 aus aufgerufen werden. „Makrofunktionsparameter“ auf Seite 27 enthält Informationen zu den Eingabeparametern der Makrofunktionen.

Zugehörige Informationen:

„Statistische Funktionen“

„Mathematische und trigonometrische Funktionen“ auf Seite 12

„Zeichenfolgefunktionen“ auf Seite 13

„Datums- und Zeitfunktionen“ auf Seite 14

„Verschiedene Funktionen“ auf Seite 14

Statistische Funktionen

Makroname	Rückgabe	Beschreibung
AVG	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das arithmetische Mittel oder den Durchschnitt eines Zellenbereichs.
MAX	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das Maximum eines Zellenbereichs.

Makroname	Rückgabe	Beschreibung
MEAN	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das arithmetische Mittel oder den Durchschnitt eines Zellenbereichs.
MIN	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das Minimum eines Zellenbereichs.
STDV oder STDEV	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Standardabweichung eines Zellenbereichs.

Mathematische und trigonometrische Funktionen

Makroname	Rückgabe	Beschreibung
AVG	Eine Spalte für jede Eingabespalte	Berechnet das arithmetische Mittel oder den Durchschnitt der Zellen im angegebenen Datenbereich.
MAX	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das Maximum eines Zellenbereichs.
MEAN	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das arithmetische Mittel oder den Durchschnitt eines Zellenbereichs.

Makroname	Rückgabe	Beschreibung
MIN	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das Minimum eines Zellenbereichs.
STDV oder STDEV	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Standardabweichung eines Zellenbereichs.
SUM	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Summe eines Zellenbereichs.
TOTAL	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Summe eines Zellenbereichs.

Zeichenfolgefunktionen

Makroname	Rückgabe	Beschreibung
LIKE	Eine Spalte für jede Eingabespalte	Ermittelt, ob eine Zeichenfolge mit einem angegebenen Muster übereinstimmt.
LOWER	Eine Spalte für jede Eingabespalte	Konvertiert Zeichenfolgewerte in Kleinschreibung.
LTRIM	Eine Spalte für jede Eingabespalte	Entfernt führende Leerzeichen aus jedem Zeichenfolgewert.
NUMBER	Eine Spalte für jede Eingabespalte	Konvertiert ASCII-Zeichenfolgen für Uhrzeit- und Datumsangaben in numerische Werte.
RTRIM	Eine Spalte für jede Eingabespalte	Entfernt nachfolgende Leerzeichen aus jedem Zeichenfolgewert.

Makroname	Rückgabe	Beschreibung
STRING_CONCAT	Eine Spalte mit einem Wert für jede Zeile der kürzesten Eingabespalte	Verkettet Zeichenfolgen aus den angegebenen Datenbereichen.
SUBSTR oder SUBSTRING	Eine Spalte für jede Eingabespalte	Gibt Zeichen aus einer Zeichenfolge ab einer Anfangsposition zurück.
UPPER	Eine Spalte für jede Eingabespalte	Konvertiert Zeichenfolgewerte in Großschreibung.

Datums- und Zeitfunktionen

Makroname	Rückgabe	Beschreibung
ADD_MONTHS	Eine Spalte für jede Eingabespalte	Gibt nach Addition der angegebenen Anzahl von Monaten ein Datum zurück.
CURRENT_DATE	Eine Spalte für jede Eingabespalte	Gibt das aktuelle Datum in einem bestimmten Format zurück.
CURRENT_DAY	Eine Spalte für jede Eingabespalte	Gibt den aktuellen Tag des Monats als Zahl zwischen 1 und 31 zurück.
CURRENT_MONTH	Eine Spalte für jede Eingabespalte	Gibt den aktuellen Monat des Jahres als Zahl zwischen 1 und 12 zurück.
CURRENT_WEEKDAY	Eine Spalte für jede Eingabespalte	Gibt den aktuellen Wochentag als Zahl zwischen 0 und 6 zurück.
CURRENT_YEAR	Eine Spalte für jede Eingabespalte	Gibt das aktuelle Jahr als Zahl zurück.
DATE	Eine Spalte für jede Eingabespalte	Konvertiert eine Datumszeichenfolge in ein julianisches Datum.
DATE_FORMAT	Eine Spalte für jede Eingabespalte	Wandelt Datumsformate um.

Verschiedene Funktionen

Makroname	Rückgabe	Beschreibung
EXTERNALCALLOUT	Werte, die von der über die ExternalCallout-API erstellten kundenspezifischen Anwendung definiert worden sind	Ruft eine kundenspezifische Anwendung auf, die mithilfe der ExternalCallout-API geschrieben wurde. Weitere Informationen finden Sie im <i>IBM Interact-Entwicklerhandbuch</i> .
IF	Eine Spalte mit einem Wert für jede Zeile der kürzesten Eingabespalte	Leitet eine If-then-else-Anweisung einer bedingten Anweisung ein.

Makrofunktionsparameter

Dieser Abschnitt beschreibt die Parameter und ihre Verwendung für Makrofunktionen in IBM Interact.

Formatspezifikationen

Dieser Abschnitt beschreibt das Format für einige häufig verwendete Parameter. Es gilt für alle Referenzen auf diese Parameter durch Makrofunktionsspezifikationen in diesem Abschnitt.

Daten

Der Parameter Daten stellt eine Datenspalte dar, auf die eine Makrofunktion angewendet werden soll. Es kann sich um eine Konstante oder um ein Feld handeln. Nähere Einzelheiten finden Sie im Abschnitt zur jeweiligen Makrofunktion.

Anmerkung: IBM Interact unterstützt keine Berechnungen in mehreren Feldern gleichzeitig oder in einem Subset von Zeilen, wie dies in IBM PredictiveInsight möglich ist.

Einige weitere Parameternamen verwenden dasselbe Format wie Daten. Die Beschreibungen dieser Parameter verweisen auf dieses Kapitel und Format.

Schlüsselwort

Der Parameter Schlüsselwort steuert das Verhalten der Makrofunktion. Er zeigt an, dass ein Schlüsselwort angegeben werden kann (wenn es weggelassen wird, wird der Standardwert verwendet). Die Schlüsselwortoptionen werden für die jeweilige Makrofunktion in der folgenden Form aufgeführt:

`{Option1 | Option2 | Option3}`

Wählen Sie die Schlüsselwortoption aus, die zu dem gewünschten Verhalten führt. Die Standardoption wird in Fettschrift angezeigt. Es können zum Beispiel die folgenden Optionen angegeben sein:

`{RADIANS | DEGREES}`

In diesem Fall sind die beiden folgenden Makrofunktionen gültig:

`COS(W1, RADIANS)` und `COS(W1, DEGREES)`

Anmerkung: Viele Makrofunktionen setzen die Schlüsselwortparameter `{ALL | COL | ROW}` ein. Diese Schlüsselwörter gelten nicht für IBM Interact, da es sich bei den Eingabedaten immer um eine Einzelspalte oder ein Einzelfeld handelt. Das Makro verhält sich immer so, als ob das Schlüsselwort COL angegeben würde. Deshalb brauchen Sie diese Schlüsselwörter nicht anzugeben, wenn Sie IBM Interact verwenden.

Verwendung von Konstanten

Die meisten Makrofunktionsparameter verarbeiten numerische Konstanten oder Ausdrücke, deren Auswertung eine numerische Konstante ergibt (Makrofunktionen für Zeichenfolgen können Zeichenfolgekonstanten verarbeiten).

In Makrofunktionen, die alle Einträge einzeln durchgehen (etwa bei der Addition zweier numerischer Spalten), ist die Verwendung einer Konstante äquivalent zur Angabe einer Spalte, die in jeder Zeile den betreffenden konstanten Wert enthält. Im Wesentlichen wird eine Konstante, die als ein Eingabeparameter angegeben ist, bis zur Länge der Eingabe expandiert.

Einige Makrofunktionen können ASCII-Zeichenfolgen und numerische Konstanten verarbeiten. Bei Parametern, die sowohl numerische Konstanten als auch ASCII-Zeichenfolgen verarbeiten können, ist dies im Abschnitt "Parameter" der jeweiligen Makrofunktion vermerkt.

Die folgende Tabelle enthält einige Beispiele.

Funktionsdefinition	Interpretation der Konstante
PERCENT_UTILIZ = (CURR_BAL*100)/ CREDIT_LIM	Die Konstante 100 wird als Spalte interpretiert, die dieselbe Zeilenzahl wie die Spalte CURR_BAL enthält, wobei jede Zeile die Konstante 100 enthält. Das abgeleitete Feld PERCENT_UTILIZ enthält jeden Wert von CURR_BAL, multipliziert mit 100 und dividiert durch jeden Wert von CREDIT_LIM.
NAME = STRING_CONCAT ("Herr ", NACHNAME)	Die Konstante "Herr " wird als Spalte interpretiert, die dieselbe Zeilenzahl wie die Spalte NACHNAME enthält, wobei jede Zeile die Konstante "Herr " enthält. Im abgeleiteten Feld NAME ist jeder der Zeichenfolgen in NACHNAME die Zeichenfolge "Herr " vorangestellt.

Anmerkung: Konstanten wie DT_BEGRENZ_M_T_J erfordern einfache Anführungszeichen.

Kapitel 3. Verwendung von Makros in IBM PredictiveInsight

Dieses Kapitel enthält Informationen zur Syntax der Makros, die in IBM PredictiveInsight zur Verfügung stehen. Alle Benutzer von IBM PredictiveInsight sollten dieses Kapitel lesen, bevor sie die übrigen Kapitel des vorliegenden Handbuchs verwenden.

Zusammenfassungen der Makrofunktionen

Die Tabellen in diesem Abschnitt fassen die Makrofunktionen nach den folgenden Kategorien zusammen:

Wichtig: Nicht alle Makros, die in diesem Handbuch aufgeführt sind, sind in IBM PredictiveInsight verfügbar. Makros, die nur in IBM Interact verfügbar sind, sind durch folgendes Symbol gekennzeichnet: **I**

Ausführliche Referenzseiten zu den einzelnen Makrofunktionen können in alphabetischer Reihenfolge von Kapitel 4, „IBM -Makroreferenz“, auf Seite 33 ausgerufen werden. „Makrofunktionsparameter“ auf Seite 27 enthält Informationen zu den Eingabeparametern der Makrofunktionen.

Zugehörige Informationen:

„Statistische Funktionen“

„Mathematische und trigonometrische Funktionen“ auf Seite 19

„Entwicklungsfunktionen“ auf Seite 22

„Zeichenfolgefunktionen“ auf Seite 23

„Datums- und Zeitfunktionen“ auf Seite 24

„Gruppierungsfunktionen“ auf Seite 8

Statistische Funktionen

Makroname	Rückgabe	Beschreibung
AVG	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das arithmetische Mittel oder den Durchschnitt eines Zellenbereichs.
AVG_DEV	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die mittlere Abweichung eines Zellenbereichs.

Makroname	Rückgabe	Beschreibung
HISTOGRAM	Einzelwert in einer neuen Spalte	Berechnet das Histogramm eines angegebenen Datenbereichs mithilfe angegebener Bin-Grenzen.
KURTOSIS	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Kurtosis eines Zellenbereichs.
MEAN	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das arithmetische Mittel oder den Durchschnitt eines Zellenbereichs.
SKEW	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Schiefe der Verteilung eines Zellenbereichs.
STAT	Variable Anzahl von Spalten (siehe Makro)	Berechnet das erste bis vierte Moment des angegebenen Datenbereichs.
STDV oder STDEV	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Standardabweichung eines Zellenbereichs.
VARIANCE	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Varianz eines Zellenbereichs.
XTAB	Eine Spalte für jeden unterschiedlichen Wert im zweiten Eingabeparameter, wobei die Anzahl der Zeilen mit der Anzahl unterschiedlicher Werte im ersten Eingabeparameter übereinstimmt	Berechnet unterschiedliche Werte in zwei Spalten und gibt COUNTOF für jede Kombination zurück.

Mathematische und trigonometrische Funktionen

Makroname	Rückgabe	Beschreibung
ABS	Eine Spalte für jede Eingabespalte	Berechnet den absoluten Wert des Inhalts des angegebenen Datenbereichs.
ACOS	Eine Spalte für jede Eingabespalte	Berechnet den Arkuskosinus des Inhalts des angegebenen Datenbereichs.
ACOT	Eine Spalte für jede Eingabespalte	Berechnet den Arkuskotangens des Inhalts des angegebenen Datenbereichs.
ASIN	Eine Spalte für jede Eingabespalte	Berechnet den Arkussinus des Inhalts des angegebenen Datenbereichs.
ATAN	Eine Spalte für jede Eingabespalte	Berechnet den Arkustangens des Inhalts des angegebenen Datenbereichs.
CEILING	Eine Spalte für jede Eingabespalte	Berechnet die Obergrenze jedes Werts im angegebenen Datenbereich.
COS	Eine Spalte für jede Eingabespalte	Berechnet den Kosinus des Inhalts des angegebenen Datenbereichs.
COSH	Eine Spalte für jede Eingabespalte	Berechnet den Hyperbelkosinus des Inhalts des angegebenen Datenbereichs.
COT	Eine Spalte für jede Eingabespalte	Berechnet den Kotangens des Inhalts des angegebenen Datenbereichs.
COUNT	Eine Spalte mit einem Einzelwert	Zählt die Anzahl der Zellen, die Werte enthalten, im angegebenen Datenbereich.
COV	Einzelwert in einer oder mehreren Spalten	Berechnet die Kovarianz von zwei Eingabebereichen.
DERIVATIVE	Eine Spalte für jede Eingabespalte	Berechnet das abgeleitete Produkt der Werte im angegebenen Datenbereich.
DIV	Eine Spalte für jede Eingabespalte	Dividiert einen angegebenen Datenbereich durch einen anderen.
EQ	Eine Spalte für jede Eingabespalte	Gibt TRUE zurück, wenn der eine Datenbereich dem anderen gleich ist.
EXP	Eine Spalte für jede Eingabespalte	Erhebt die natürliche Zahl (e) in die Potenz, die durch den Inhalt jeder Zelle im angegebenen Datenbereich angegeben ist.
FACTORIAL	Eine Spalte für jede Eingabespalte	Berechnet die Fakultät jedes Werts im angegebenen Datenbereich.

Makroname	Rückgabe	Beschreibung
FLOOR	Eine Spalte für jede Eingabespalte	Berechnet die Untergrenze jedes Werts im angegebenen Datenbereich.
FRACTION	Eine Spalte für jede Eingabespalte	Berechnet die Nachkommastellen jedes Werts im angegebenen Datenbereich.
GAUSS	Eine Spalte für jede Eingabespalte	Berechnet die gaußsche Kurve der Werte im angegebenen Datenbereich.
GAUSS_AREA	Eine Spalte für jede Eingabespalte	Berechnet die Fläche unter der gaußschen Kurve der Werte im angegebenen Datenbereich.
GE	Eine Spalte für jede Eingabespalte	Gibt TRUE zurück, wenn der eine Datenbereich größer-gleich dem anderen ist.
GT	Eine Spalte für jede Eingabespalte	Gibt TRUE zurück, wenn der eine Datenbereich größer als der andere ist.
INT	Eine Spalte für jede Eingabespalte	Berechnet den abgerundeten Ganzzahlwert des Inhalts des angegebenen Datenbereichs.
INTEGRAL	Eine Spalte für jede Eingabespalte	Berechnet das Integral der Werte im angegebenen Datenbereich.
INVERSE	Eine Spalte für jede Eingabespalte	Berechnet den Negativwert des Inhalts des angegebenen Datenbereichs.
ISEVEN	Eine Spalte für jede Eingabespalte	Prüft, ob die Eingabewerte gerade (d. h. durch zwei teilbar) sind.
ISODD	Eine Spalte für jede Eingabespalte	Prüft, ob die Eingabewerte ungerade (d. h. nicht durch zwei teilbar) sind.
LE	Eine Spalte für jede Eingabespalte	Gibt TRUE zurück, wenn der eine Datenbereich kleiner-gleich dem anderen ist.
LN	Eine Spalte für jede Eingabespalte	Berechnet den natürlichen Logarithmus des Inhalts des angegebenen Datenbereichs.
LOG	Eine Spalte für jede Eingabespalte	Berechnet den natürlichen Logarithmus des Inhalts des angegebenen Datenbereichs.
LOG2	Eine Spalte für jede Eingabespalte	Berechnet den binären Logarithmus des Inhalts des angegebenen Datenbereichs.
LOG10	Eine Spalte für jede Eingabespalte	Berechnet den dekadischen Logarithmus des Inhalts des angegebenen Datenbereichs.

Makroname	Rückgabe	Beschreibung
LT	Eine Spalte für jede Eingabespalte	Gibt TRUE zurück, wenn der eine Datenbereich kleiner als der andere ist.
MAX	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das Maximum eines Zellenbereichs.
MIN	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet das Minimum eines Zellenbereichs.
MINUS	Eine Spalte für jede Eingabespalte	Subtrahiert einen angegebenen Datenbereich von einem anderen.
MOD	Eine Spalte für jede Eingabespalte	Berechnet den Modulo-Wert des Inhalts des angegebenen Datenbereichs.
MULT	Eine Spalte für jede Eingabespalte	Multipliziert die Inhalte zweier Datenbereiche.
NE	Eine Spalte für jede Eingabespalte	Gibt TRUE zurück, wenn der eine Datenbereich dem anderen ungleich ist.
PLUS	Eine Spalte für jede Eingabespalte	Addiert die Inhalte zweier Datenbereiche.
POW	Eine Spalte für jede Eingabespalte	Erhebt einen Basiswert in die Potenz des angegebenen Exponenten.
RANDOM	Eine Spalte mit der angegebenen Anzahl von Werten	Gibt die angegebene Anzahl von Zufallszahlen zurück.
RANDOM_GAUSS	Eine Spalte mit der angegebenen Anzahl von Werten	Gibt die angegebene Anzahl von Zufallswerten aus einer gaußschen Verteilung zurück.
ROUND	Eine Spalte für jede Eingabespalte	Berechnet den gerundeten Wert des Inhalts des angegebenen Datenbereichs.
SIGN	Eine Spalte für jede Eingabespalte	Berechnet das Vorzeichen (positiv oder negativ) der Werte im angegebenen Datenbereich.
SIN	Eine Spalte für jede Eingabespalte	Berechnet den Sinus des Inhalts des angegebenen Datenbereichs.

Makroname	Rückgabe	Beschreibung
SINH	Eine Spalte für jede Eingabespalte	Berechnet den Hyperbelsinus des Inhalts des angegebenen Datenbereichs.
SQRT	Eine Spalte für jede Eingabespalte	Berechnet die Quadratwurzel des Inhalts des angegebenen Datenbereichs.
SUM	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Summe eines Zellenbereichs.
TAN	Eine Spalte für jede Eingabespalte	Berechnet den Tangens des Inhalts des angegebenen Datenbereichs.
TANH	Eine Spalte für jede Eingabespalte	Berechnet den Hyperbeltangens des Inhalts des angegebenen Datenbereichs.
TOTAL	Bei dem Schlüsselwort ALL ein Einzelwert in einer neuen Spalte; bei dem Schlüsselwort COL eine Spalte mit einem Einzelwert für jede Eingabespalte; bei dem Schlüsselwort ROW eine Spalte mit einem Wert für jede Zeile	Berechnet die Summe eines Zellenbereichs.
TRUNCATE	Eine Spalte für jede Eingabespalte	Berechnet den Ganzzahlanteil jedes Werts im angegebenen Datenbereich.

Entwicklungsfunktionen

Makroname	Rückgabe	Funktion
DELAY	Eine Spalte für jede Eingabespalte	Gibt die Verzögerung der Eingabespaltenwerte um eine angegebene Anzahl von Zeitschritten an.
GRID	Eine Spalte für jede Eingabespalte	Gibt ein Grid aller möglichen Wertekombinationen (eine pro Zeile) zurück.
LAG	Eine Spalte für jede Eingabespalte	Gibt die Verzögerung der Eingabespaltenwerte um eine angegebene Anzahl von Zeitschritten an.
NORM_MAXMIN	Eine Spalte für jede Eingabespalte	Berechnet die Minimum/Maximum-Normalisierung eines Datenbereichs.

Makroname	Rückgabe	Funktion
NORM_SIGMOID	Eine Spalte für jede Eingabespalte	Berechnet die sigmoidale Normalisierung eines Datenbereichs.
NORM_ZSCORE	Eine Spalte für jede Eingabespalte	Berechnet die z-Faktorwert-Normalisierung eines Datenbereichs.
PCA	Eine Spalte für jede Eingabespalte sowie eine weitere Spalte	Berechnet die Eigenvektoren für Hauptkomponenten des angegebenen Datenbereichs.
PCA_FEATURES	Eine Spalte für jede Eingabespalte sowie eine weitere Spalte	Extrahiert n Komponenten aus dem angegebenen Datenbereich.
SAMPLE_RANDOM	Eine Spalte für jede Eingabespalte	Gibt Spalte(n) von n Zellen zurück, die jeweils eine Zufallsstichprobe aus dem angegebenen Datenbereich enthalten.
SLIDE_WINDOW	Anzahl der Eingabespalten multipliziert mit dem Parameter Breite	Erstellt ein Muster aus einem angegebenen Fenster und schiebt es nach unten, um das nächste Muster zu erstellen.
SORT	Eine Spalte für jede Eingabespalte	Sortiert die Werte im angegebenen Datenbereich in aufsteigender oder absteigender Reihenfolge.
SUBSAMPLE	Eine Spalte für jede Eingabespalte	Reduziert Daten durch Rückgabe jedes n-ten Zeilenwerts.

Zeichenfolgelfunktionen

Makroname	Rückgabe	Funktion
DISTINCT	Eine Spalte für jede Eingabespalte	
FORMAT	Eine Spalte für jede Eingabespalte	Steuert die Ausgabeformatierung für Zahlen und Zeichenfolgen (z. B. Ausgabebreite, Ausrichtung, numerische Genauigkeit, Dezimalzeichen, Trennzeichen usw.). Gibt die formatierte Ausgabezeichenfolge zurück.
LIKE	Eine Spalte für jede Eingabespalte	Ermittelt, ob eine Zeichenfolge mit einem angegebenen Muster übereinstimmt.
NUMBER	Eine Spalte für jede Eingabespalte	Konvertiert ASCII-Zeichenfolgen für Uhrzeit- und Datumsangaben in numerische Werte.

Makroname	Rückgabe	Funktion
POSITION	Eine Spalte für jede Eingabespalte	Gibt die Anfangsposition eines Musters in einer Zeichenfolge zurück.
STRING_CONCAT	Eine Spalte mit einem Wert für jede Zeile der kürzesten Eingabespalte	Verkettet Zeichenfolgen aus den angegebenen Datenbereichen.
STRING_HEAD	Eine Spalte für jede Eingabespalte	Gibt die ersten n Zeichen jeder Zeichenfolge im angegebenen Datenbereich zurück.
STRING_LENGTH	Eine Spalte für jede Eingabespalte	Gibt die Länge jeder Zeichenfolge im angegebenen Datenbereich zurück.
STRING_SEG	Eine Spalte für jede Eingabespalte	Gibt das Zeichenfolgesegment zwischen zwei angegebenen Indizes zurück.
STRING_TAIL	Eine Spalte für jede Eingabespalte	Gibt die letzten n Zeichen jeder Zeichenfolge im angegebenen Datenbereich zurück.
SUBSTR	Eine Spalte für jede Eingabespalte	Gibt Zeichen aus einer Zeichenfolge ab einer Anfangsposition zurück.
SUBSTRING	Eine Spalte für jede Eingabespalte	Gibt Zeichen aus einer Zeichenfolge ab einer Anfangsposition zurück.

Datums- und Zeitfunktionen

Makroname	Rückgabe	Beschreibung
ADD_MONTHS	Eine Spalte für jede Eingabespalte	Gibt nach Addition der angegebenen Anzahl von Monaten ein Datum zurück.
CURRENT_DATE	Eine Spalte für jede Eingabespalte	Gibt das aktuelle Datum in einem bestimmten Format zurück.
CURRENT_DAY	Eine Spalte für jede Eingabespalte	Gibt den aktuellen Tag des Monats als Zahl zwischen 1 und 31 zurück.
CURRENT_JULIAN	Eine Spalte für jede Eingabespalte	Gibt die julianische Zahl für das aktuelle Datum zurück.
CURRENT_MONTH	Eine Spalte für jede Eingabespalte	Gibt den aktuellen Monat des Jahres als Zahl zwischen 1 und 12 zurück.
CURRENT_TIME	Eine Spalte für jede Eingabespalte	Gibt die aktuelle Uhrzeit als Zeichenfolge zurück.
CURRENT_WEEKDAY	Eine Spalte für jede Eingabespalte	Gibt den aktuellen Wochentag als Zahl zwischen 0 und 6 zurück.
CURRENT_YEAR	Eine Spalte für jede Eingabespalte	Gibt das aktuelle Jahr als Zahl zurück.

Makroname	Rückgabe	Beschreibung
DATE	Eine Spalte für jede Eingabespalte	Konvertiert eine Datumszeichenfolge in ein julianisches Datum.
DATE_FORMAT	Eine Spalte für jede Eingabespalte	Wandelt Datumsformate um.
MONTHOF	Eine Spalte für jede Eingabespalte	Gibt den Monat des Jahres als Zahl zurück.
WEEKDAY	Eine Spalte für jede Eingabespalte	Konvertiert ASCII-Datumszeichenfolgen in Wochentage.
WEEKDAYOF	Eine Spalte für jede Eingabespalte	Gibt den Wochentag der Woche als Zahl zurück.
YEAROF	Eine Spalte für jede Eingabespalte	Gibt das Jahr als Zahl zurück.

Gruppierungsfunktionen

Makroname	Rückgabe	Beschreibung
GROUPBY	Eine neue Spalte mit einem Wert für jede Zeile	Fasst mehrere Zeilen von Daten in einer Gruppe zusammen.
GROUPBY_WHERE	Eine neue Spalte mit einem Wert für jede Zeile	Fasst mehrere Zeilen von Daten zusammen, die eine bestimmte Bedingung erfüllen und sich in einer Gruppe befinden.

Verschiedene Funktionen

Makroname	Rückgabe	Funktion
BUFFER	Eine Spalte für jede Eingabespalte	Kopiert den Eingabedatenbereich und aktualisiert die Werte dynamisch.
COLUMN	Eine Spalte für jede Eingabespalte	Erstellt neue Spalten, wobei die Eingabewerte in jeder Spalte vertikal verkettet werden
CONSTANT	Eine Spalte für jede Eingabespalte	Kopiert den Eingabedatenbereich einmal (keine dynamische Aktualisierung)
COUNT_DIFF	Zwei Spalten	Gibt jeden eindeutigen Wert in der Eingabe mit einer Zahl zurück, die die Häufigkeit des betreffenden Werts angibt.
CV_FOLDS	Eine Spalte mit einem Wert für jede Zeile der kürzesten Eingabespalte	Unterteilt Eingabemuster in n Teilmengen von Daten zur Kreuzvalidierung

Makroname	Rückgabe	Funktion
DATALINK	Eine Spalte für jede Spalte mit verknüpften Daten	Erstellt einen internen Link zu Daten in einem IBM PredictiveInsight-Arbeitsblatt
DDELINK	Eine Spalte für jede Spalte mit verknüpften Daten	Erstellt einen externen Link zu Daten aus einer anderen Windows-Anwendung.
DECIMATE	MAX_WERT Spalten (eine Spalte für jeden Ausgabewert)	Wandelt eine Spalte von Zahlen in mehrere Spalten um, bei denen eine Eins den Indexwert angibt.
EXTRACT	Eine Spalte für jede Eingabespalte	Extrahiert Zeilen aufgrund der Werte in einer Prädikatspalte.
IF	Eine Spalte mit einem Wert für jede Zeile der kürzesten Eingabespalte	Leitet eine If-then-else-Anweisung einer bedingten Aussage ein.
INIT	Eine Spalte	Initialisiert Werte vorheriger Zeitschritte für eine rekursive Funktion
ISERROR	Eine Spalte mit einem Wert für jede Zeile der kürzesten Eingabespalte	Gibt eine Eins zurück, wenn ein Wert in der Eingabezeile eine fehlerhafte Zelle (???) enthält, andernfalls eine Null
ISMEMBER	Eine Spalte für jede Eingabespalte	Prüft einen Eingabebereich anhand einer "Tabelle" von Werten und gibt eine Eins zurück, wenn ein Wert in der Tabelle enthalten ist, andernfalls eine Null.
MAXINDEX	Eine Spalte mit einem Wert für jede Zeile der kürzesten Eingabespalte	Gibt den Spaltenindex des n-ten (ersten, zweiten, dritten usw.) Maximalwerts für jede Zeile der angegebenen Spalte zurück
RANK	Eine Spalte für jede Eingabespalte	Unterteilt Daten in n-Bins (Standard 10) Gruppen, die jeweils die etwa gleiche Anzahl unterschiedlicher Werte enthalten, und gibt die Gruppe zurück, der die jeweilige Datenzeile angehört
SORT	Eine Spalte für jede Eingabespalte	Sortiert den angegebenen Datenbereich in aufsteigender oder absteigender Reihenfolge

Hinweise zu Makroreferenzen

Jede der verfügbaren Makrofunktionen wird in dem Abschnitt *Zusammenfassungen der Makrofunktionen* für das jeweilige Produkt beschrieben. Lesen Sie die Syntaxkonventionen im Vorwort durch, bevor Sie beginnen, mit einem Makro zu arbeiten.

Für jede Makrofunktion sind einfache Beispiele angegeben. In diesen Beispielen erhalten zusätzliche Spalten die Namen WX, WY, WZ usw. Welche Namen tatsächlich in Ihrem Arbeitsblatt verwendet werden, hängt von der konkreten Situation ab.

Makrofunktionsparameter

Dieser Abschnitt beschreibt die Parameter und ihre Verwendung für Makrofunktionen in IBM PredictiveInsight.

Formatspezifikationen

Dieser Abschnitt beschreibt das Format für einige häufig verwendete Parameter. Es gilt für alle Referenzen auf diese Parameter durch Makrofunktionsspezifikationen in diesem Kapitel.

Daten

Der Parameter Daten stellt einen Datenbereich dar, auf den eine Makrofunktion angewendet werden soll. Es handelt sich in der Regel um eine Konstante, eine Spalte oder einen Zellenbereich (nähere Informationen finden Sie im Abschnitt zur jeweiligen Makrofunktion). Das Format für den Parameter Daten lautet wie folgt:

Anfangsdaten [: Enddaten]

Bei Anfangsdaten kann es sich um eine Konstante (z. B. 10,2), den Namen einer Spalte (z. B. W1) oder einen Zellenbereich (z. B. W1[1:100]) handeln. Der Parameter Enddaten ist optional. Wenn er angegeben wird, dienen die Anfangsdaten als Ausgangspunkt und müssen eine Spalte oder einen Zellenbereich darstellen. Der Endpunkt wird durch Enddaten angegeben.

Anmerkung: Einige weitere Parameternamen verwenden dasselbe Format wie Daten. Die Beschreibungen dieser Parameter verweisen auf dieses Kapitel und Format.

Schlüsselwort

Der Parameter Schlüsselwort steuert das Verhalten der Makrofunktion. Er zeigt an, dass ein Schlüsselwort angegeben werden kann (wenn es weggelassen wird, wird der Standardwert verwendet). Die Schlüsselwortoptionen werden für die jeweilige Makrofunktion in der folgenden Form aufgeführt:

{**Option1** | Option2 | Option3}

Wählen Sie die Schlüsselwortoption aus, die zu dem gewünschten Verhalten führt. Die Standardoption wird in Fettschrift angezeigt. Es kann zum Beispiel das folgende Format angegeben sein:

{**ALL** | COL | ROW}

In diesem Fall sind alle folgenden Makrofunktionen gültig:

AVG(W1:W5) AVG(W1:W5, ALL) AVG(W1:W5, COL) AVG(W1:W5, ROW)

Verwendung von Zellen- und Spaltenbereichen

Bei den meisten Parametern von Makrofunktionen können Zellen- und Spaltenbereiche als Eingaben angegeben werden. Sie müssen die folgenden Regeln einhalten:

- Die Anzahl der Spalten muss übereinstimmen.
- Die Rückgabewerte beginnen in der ersten Zelle.

- Zellenbereiche werden automatisch mit Nullen gefüllt.

Anzahl der Spalten muss übereinstimmen

Wenn zwei oder mehr Datenbereiche als Eingabe angegeben sind und die Berechnungen spaltenweise ausgeführt werden, müssen die beiden Datenbereiche dieselbe Spaltenanzahl aufweisen. Andernfalls werden nur die Dimensionen des kleineren Datenbereichs verwendet (einige Makrofunktionen melden einen Fehler). Wenn die Datenbereiche unterschiedlich viele Zeilen enthalten, führen die meisten Makrofunktionen die Berechnungen bis zur letzten Zeile der kürzesten Spalte (einschließlich) aus.

- Was zum Beispiel Spaltenbereiche betrifft, so generiert die Makrodefinition $W6 = W1:W3 \text{ AND } W4:W6$ drei Ausgabespalten (beide Datenbereiche enthalten drei Spalten). Spalte $W1$ wird durch AND mit Spalte $W4$ verknüpft, Spalte $W2$ wird durch AND mit Spalte $W5$ verknüpft und Spalte $W3$ wird durch AND mit Spalte $W6$ verknüpft. Dagegen werden bei der Definition $W6 = W1:W3 \text{ AND } W4:W5$ nur zwei Ausgabespalten zurückgegeben (der erste Spaltenbereich enthält drei Spalten, der zweite aber nur zwei; der kleinere der beiden wird verwendet). In diesem Fall wird Spalte $W1$ durch AND mit Spalte $W4$ verknüpft, Spalte $W2$ mit Spalte $W5$. Spalte $W3$ wird nicht verwendet.
- Was Zellenbereiche betrifft, so generiert die Makrodefinition $W7 = W1[1:5]:W2 \text{ AND } W4[10:50]:W5$ zwei Ausgabespalten (beide Eingabebereiche enthalten zwei Spalten). Die Ausgabespalten $W7$ und $W8$ enthalten fünf Werte (die Zellen 1-5 werden durch AND mit den Zellen 10-14 verknüpft). Die Makrodefinition $W7 = W1[1:5]:W2 \text{ AND } W4$ generiert nur eine Ausgabespalte, da der zweite Datenbereich nur eine Spalte enthält.

Wichtig: Die Angabe einer Spalte ohne einen Zellenbereich ist äquivalent zur Angabe der vollständigen Spalte (d. h. des Zellenbereichs eins für die gesamte Länge der Spalte).

Rückgabewerte beginnen in der ersten Zelle

Von einer Makrofunktion Rückgabewerte werden in fortlaufenden Zellen angeordnet, beginnend mit der ersten Zelle (z. B. $TEMP[1]$). So würden bei der Berechnung von $W2 = SIN(W1[100:200])$ 101 Werte in Zelle 1-100 von Spalte $W2$ angeordnet.

Wichtig: Wenn Sie eine Operation zeilenweise auf einen Zellenbereich anwenden müssen und die Ergebnisse in den entsprechenden Zeilen behalten möchten (d. h., wenn Sie die Zellen $[10:20]$ bearbeiten und die Ergebnisse in Zelle 10-20 der Ausgabespalte vorfinden möchten), müssen Sie den Zellenbereich $[1:20]$ angeben. Dabei werden einige überflüssige Werte berechnet, aber die Ergebnisse in den gewünschten Zeilen angeordnet.

Automatisch mit Nullen gefüllte Zellenbereiche

Wenn Sie einen Zellenbereich angeben, werden leere (freie) Zellen in dem Zellenbereich automatisch mit Nullen gefüllt. So liefert $W3 = W1[1:3]*W2$ folgendes Ergebnis:

```
W1 W2 W3 1 2 2 3 4 12 [] 6 0
```

Dabei stellt $[]$ eine leere Zelle dar (d. h., Spalte $W1$ enthält nur zwei Zellenwerte). Bei Angabe von $W3 = W1*W2$ ergeben sich jedoch die beiden Werte 2 und 12 (Berechnungen werden bis zur kürzeren der beiden Spalten ausgeführt).

Verwendung von Konstanten

Die meisten Makrofunktionsparameter verarbeiten numerische Konstanten oder Ausdrücke, deren Auswertung eine numerische Konstante ergibt (Makrofunktionen

für Zeichenfolgen können Zeichenfolgekonstanten verarbeiten). In Makrofunktionen, die alle Einträge einzeln durchgehen, ist die Verwendung einer Konstante äquivalent zur Angabe einer Spalte, die in jeder Zeile den betreffenden konstanten Wert enthält. Im Wesentlichen wird, wenn eine Konstante und ein Zellen- oder Spaltenbereich als Eingabeparameter angegeben sind, die Konstante bis zu den Dimensionen des Zellen- oder Spaltenbereichs expandiert. Eine Spalte, die eine einzelne Zelle enthält und in einer Makrofunktion als Eingabe verwendet wird, wird als Konstante interpretiert.

Einige Makrofunktionen können ASCII-Zeichenfolgen und numerische Konstanten verarbeiten. Bei Parametern, die sowohl numerische Konstanten als auch ASCII-Zeichenfolgen verarbeiten können, ist dies im Abschnitt "Parameter" der jeweiligen Makrofunktion vermerkt.

Die folgende Tabelle enthält einige Beispiele.

Funktionsdefinition	Interpretation der Konstante
$W1 = 3+5$	Jede der Konstanten wird als einzelne Spalte mit dem einzelnen Wert interpretiert. Die Spalte W1 enthält daher den Einzelwert 8.
$W2 = 2*W1$	Die Konstante 2 wird als Spalte interpretiert, die dieselbe Zeilenzahl wie die Spalte W1 enthält, wobei jede Zeile die Konstante 2 enthält. Die Spalte W2 enthält daher jeden Wert von W1, multipliziert mit 2.
$W2 = \text{STRING_CONCAT}(W1, "in")$	Die Konstante "in" wird als Spalte interpretiert, die dieselbe Zeilenzahl wie die Spalte W1 enthält, wobei jede Zeile die Konstante "in" enthält. In der Spalte W2 ist jede der Zeichenfolgen in W1 mit der Zeichenfolge "in" verkettet.
$W4 = W1:W3 / \text{AVG}(W1:W3)$	Die Auswertung des Ausdrucks $\text{AVG}(W1:W3)$ ergibt einen konstanten Wert x . Die Konstante x wird als drei Spalten interpretiert, die so viele Zeilen wie die kürzeste Spalte von W1, W2 oder W3 enthalten. Jede Zelle enthält die Konstante x . Die Ausgabespalten W4-W6 enthalten die Werte aus den Spalten W1-W3, die durch x dividiert sind.
$W3 = W1[10:20]^2$	Die Konstante 2 wird als Zellenbereich mit 11 Zeilen interpretiert, die jeweils den Wert 2 enthalten.

Rechengenauigkeit

Alle Berechnungen in **IBM PredictiveInsight**-Arbeitsblättern sind auf eine maximale Genauigkeit von 32 Bit begrenzt.

Ganzzahlberechnungen

Makrofunktionen, die Ganzzahlberechnungen ausführen (BIT_AND, BIT_NOT, BIT_OR, BIT_XOR und T0), verarbeiten keine negativen Zahlen. Die Werte müssen zwischen 0 und $(2^{24} - 1)$ liegen; andernfalls wird ein Fehler zurückgegeben.

Ergebnisse ungültiger Zellen und leere Zellen

Zellen mit ???

Wenn eine Arbeitsblattoperation ungültige Ergebnisse erzeugt hat, enthält die Zelle ??? statt eines berechneten Werts. Wenn beispielsweise durch die Makrofunktion SQRT die Quadratwurzel einer negativen Zahl gezogen wird, wird für jeden negativen Eingabewert ??? generiert.

Berechnungen aufgrund von ???

Wenn eine Zelle den Wert ??? erhalten hat, geben die meisten Berechnungen, die diese Zelle verwenden, das Ergebnis ??? weiter. Wenn beispielsweise für eine Spalte, die eine oder mehrere Zellen mit ??? enthält, die Summe gebildet wird, lautet das Ergebnis ???.

Wenn in einer oder mehreren Zellen das Ergebnis ??? auftritt, dann klicken Sie auf eine der Zellen mit ??? und lesen Sie die Fehlernachricht, die im Textfenster **Funktionsdefinition** angezeigt wird. Die Makrofunktionsreferenz im vorliegenden Handbuch enthält Informationen zur Ursache des Fehlers. Ändern Sie die Funktionsdefinition so, dass alle Eingabewerte gültig sind. So könnten Sie bei dem Beispiel mit der Quadratwurzel zuerst den absoluten Wert der Eingabedaten ermitteln:

$W2 = \text{SQRT}(W1) \rightarrow W2 = \text{SQRT}(\text{ABS}(W1))$

Anmerkung: Unter Umständen müssen Sie ???-Werte in abhängigen Spalten zurückverfolgen, um das Problem zu beheben.

Wichtig: Alle Zellen mit ???, die als Teil eines Trainings- oder Testmusters an den Experimentmanager übergeben werden, werden als Nullen übergeben.

Leere Zellen und ???-Zellen

Leere Zellen sind einfach Zellen ohne Inhalt. Sie treten nur am Ende einer Spalte oder in leeren Spalten auf. Leere Zellen und ???-Zellen werden von verschiedenen Makrofunktionen unterschiedlich behandelt. Die meisten Makrofunktionen behandeln leere Zellen als Nullen, während sie bei ???-Zellen Fehler auslösen und weitergeben. Die Ausnahmen sind folgende:

- Die Makrofunktion MAXINDEX ignoriert sowohl leere Zellen als auch ???-Zellen.
- Die Makrofunktion OFFSET ignoriert leere Zellen und ???-Zellen.
- Die Makrofunktion DDELINK gibt leere Zellen und ???-Zellen weiter.
- Informationen zu Makrofunktionen, die leere Zellen überspringen, finden Sie im Abschnitt „Verwendung von Konstanten“ auf Seite 28. Fehler in ???-Zellen werden dennoch ausgelöst und weitergegeben.
- Informationen zu Makrofunktionen, die leere Zellen überspringen, aber bei ???-Zellen wie angegeben verfahren, finden Sie im Abschnitt „Makrofunktionen, die leere Zellen überspringen und ???-Zellen verwenden“ auf Seite 31.
- Informationen zu Makrofunktionen, die für alle Zellen ??? zurückgeben, wenn eine der Zellen in den aufgelisteten Argumenten ??? enthält, finden Sie im Abschnitt „Makrofunktionen, die keine ???-Zellen verarbeiten können“ auf Seite 31.

Makrofunktionen, die leere Zellen überspringen und ???-Zellen weitergeben

AVG	NORM_MINMAX
DERIVATIVE	NORM_ZSCORE

KURTOSIS	PCA
MAX	PCA_FEATURES
MEAN	SKEW
MIN	SUM
MOMENTS	VARIANCE

Makrofunktionen, die leere Zellen überspringen und ???-Zellen verwenden

Makrofunktion	Verwendung von ???-Zellen
COUNT	??? -Zellen werden gezählt.
COLUMN	??? -Zellen werden kopiert.
CV_FOLDS	Fragezeichen (???) werden als eigene Klasse behandelt.
DELAY	??? -Zellen werden kopiert.
FREQ	Fragezeichen (???) in einer Zeichenfolgespalte werden als eigene Klasse gezählt. In einer numerischen Spalte wird ??? als eine Null gezählt.
EXTRACT	Fragezeichen (???) in der Prädikatspalte werden als Nullen behandelt.
LAG	??? -Zellen werden kopiert.
MERGE	??? -Zellen werden kopiert.
SORT	Alle ???-Zellen werden am Ende einsortiert.
SAMPLE_RANDOM	??? -Zellen ermöglichen Stichproben.
SELECT	??? -Zellen werden kopiert.
SUBSAMPLE	??? -Zellen ermöglichen Stichproben.

Makrofunktionen, die keine ???-Zellen verarbeiten können

Makrofunktion	Argument, das keine ??? enthalten darf
GRID	Spalte1, Spalte2
HISTOGRAM	Daten, Bin-Spalte
INTEGRAL	Daten, Multiplikator
ISERROR	Daten

Kapitel 4. IBM -Makroreferenz

Dieser Abschnitt enthält Referenzinformationen zu den verfügbaren Arbeitsblattmakrofunktionen in der Suite IBM Marketing Platform.

Die verfügbaren Makros sind auf den folgenden Seiten in alphabetischer Reihenfolge aufgeführt. Zu jedem Makro werden die Syntax, die möglichen Parameter sowie einige Beispiele angegeben.

Wichtig: Verwenden Sie keine Funktionsnamen oder Schlüsselwörter aus der IBM-Makrosprache für Spaltenüberschriften von Benutzertabellen in IBM Kampagne, unabhängig davon, ob deren Zuordnung auf einer Datenbank oder einer unstrukturierten Datei beruht. Diese reservierten Wörter können Fehler verursachen, wenn sie in Spaltenüberschriften von zugeordneten Tabellen verwendet werden.

ABS

Syntax

ABS(Daten)

Parameter

Daten

Die numerischen Werte, deren absoluter Wert berechnet werden soll. Dieser Parameter kann ein konstanter Wert, eine Spalte, ein Zellenbereich oder ein Ausdruck sein, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

ABS berechnet den absoluten Wert der Zahlen im angegebenen Datenbereich. Der absolute Wert einer Zahl ist ihr Wert ohne das Vorzeichen (d. h., positive Zahlen werden nicht geändert, negative Zahlen werden als positive Zahlen zurückgegeben). ABS gibt für jede Eingabespalte eine neue Spalte zurück, die den absoluten Wert der Zahlen in der entsprechenden Eingabespalte enthält.

Beispiele

TEMP = ABS(-3) oder TEMP = ABS(3)
Erstellt eine Spalte TEMP, die den Wert 3 enthält.
TEMP = ABS(W1)
Erstellt eine Spalte TEMP, in der jeder Wert den absoluten Wert des Inhalts von Spalte W1 darstellt.
TEMP = ABS(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die absoluten Werte des Inhalts von Spalte W1, die Werte von Spalte WX sind die absoluten Werte des Inhalts von Spalte W2 und die Werte von Spalte WY sind die absoluten Werte des Inhalts von Spalte W3.

TEMP = ABS(W1[10:20])

Erstellt eine Spalte TEMP, in der die ersten 11 Zellen die absoluten Werte der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.

TEMP = ABS(W1[1:5]:W2)

Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind die absoluten Werte der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind die absoluten Werte der entsprechenden Zeilen von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
SIGN	Berechnet das Vorzeichen (positiv oder negativ) der Werte im angegebenen Datenbereich.

ACOS

Syntax

ACOS(Daten [, Einheit])

Parameter

Daten

Die numerischen Werte, deren Arkuskosinus berechnet werden soll. Dieser Parameter kann ein konstanter Wert, eine Spalte, ein Zellenbereich oder ein Ausdruck sein, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Einheit

Dieses optionale Schlüsselwort legt fest, ob die Eingabewerte und Ergebnisse als Grad oder als Radiant interpretiert werden. Wählen Sie einen der folgenden Werte aus:

RADIAN - Führt die Berechnungen in Radiant aus (Standard)

DEGREE - Führt die Berechnungen in Grad aus

Wird dieser Parameter nicht angegeben, ist der Standard Radiant. (Die Umrechnung von Radiant in Grad erfolgt durch Division durch Pi und Multiplikation mit 180.)

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

ACOS berechnet den Arkuskosinus der Werte im angegebenen Datenbereich. Der Arkuskosinus ist der Winkel, dessen Kosinus der Inhalt der jeweiligen Zelle ist. ACOS gibt für jede Eingabespalte eine neue Spalte zurück, die den Arkuskosinus der Zahlen in der entsprechenden Eingabespalte enthält.

Wenn das Schlüsselwort RADIAN verwendet wird, gibt ACOS Werte im Bereich 0 bis π zurück. Wenn das Schlüsselwort DEGREE verwendet wird, gibt ACOS Werte im Bereich 0 bis 180 zurück.

Anmerkung: Der Zelleninhalt jeder angegebenen Spalte muss Werte zwischen -1,0 und 1,0 (einschließlich) aufweisen. Andernfalls wird für jede ungültige Eingabe eine leere Zelle zurückgegeben.

Beispiele

TEMP = ACOS(0) oder TEMP = ACOS(0, 0) oder TEMP = ACOS(0, RADIAN)
Erstellt eine Spalte TEMP, die den Wert 1,571 ($\pi/2$ Radiant) enthält.
TEMP = ACOS(0, 1) oder TEMP = ACOS(0, DEGREE)
Erstellt eine Spalte TEMP, die den Wert 90 (Grad) enthält.
TEMP = ACOS(W1)
Erstellt eine Spalte TEMP, in der jeder Wert den Arkuskosinus (in Radiant) des Inhalts von Spalte W1 darstellt.
TEMP = ACOS(W1:W3, 1)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der Arkuskosinus des Inhalts von Spalte W1, die Werte von Spalte WX sind der Arkuskosinus des Inhalts von Spalte W2 und die Werte von Spalte WY sind der Arkuskosinus des Inhalts von Spalte W3. Alle Werte sind in Grad angegeben.
TEMP = ACOS(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils den Arkuskosinus der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = ACOS(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der Arkuskosinus der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind der Arkuskosinus der entsprechenden Zeilen von Spalte W2. Alle Werte sind in Radiant angegeben.

Zugehörige Funktionen

Funktion	Beschreibung
ACOT	Berechnet den Arkuskotangens des Inhalts des angegebenen Datenbereichs.
ASIN	Berechnet den Arkussinus des Inhalts des angegebenen Datenbereichs.
ATAN	Berechnet den Arkustangens des Inhalts des angegebenen Datenbereichs.
COS	Berechnet den Kosinus des Inhalts des angegebenen Datenbereichs.

ACOT

Syntax

ACOT(Daten [, Einheit])

Parameter

Daten

Die numerischen Werte, deren Arkuskotangens berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Einheit

Dieses optionale Schlüsselwort legt fest, ob die Eingabewerte und Ergebnisse als Grad oder als Radiant interpretiert werden. Wählen Sie eine der folgenden Angaben aus:

RADIAN - Führt die Berechnungen in Radiant aus (Standard)

DEGREE - Führt die Berechnungen in Grad aus

Wird dieser Parameter nicht angegeben, ist der Standard Radiant. (Die Umrechnung von Radiant in Grad erfolgt durch Division durch Pi und Multiplikation mit 180.)

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

ACOT gibt den Winkel zurück, dessen Kotangens der Inhalt der jeweiligen Zelle ist. ACOT gibt für jede Eingabespalte eine neue Spalte zurück, die den Arkuskotangens der Zahlen in der entsprechenden Eingabespalte enthält. Dabei werden 64-Bit-Gleitkommazahlen verwendet.

Beispiele

TEMP = ACOT(0,5) oder TEMP = ACOT(0,5, 0) oder TEMP = ACOT(0,5, RADIAN)
Erstellt eine Spalte TEMP, die den Wert 2,157 (Radiant) enthält.
TEMP = ACOT(1, 1) oder TEMP = ACOT(1, DEGREE)
Erstellt eine Spalte TEMP, die den Wert 0,022 (1/45) Grad enthält.
TEMP = ACOT(0)
Erstellt eine Spalte TEMP, die den Wert MAX32_Float in Radiant enthält.

TEMP = ACOT(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert den Arkuskotangens (in Radiant) des Inhalts von Spalte W1 darstellt.
TEMP = ACOT(W1:W3, 1)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der Arkuskotangens des Inhalts von Spalte W1, die Werte von Spalte WX sind der Arkuskotangens des Inhalts von Spalte W2 und die Werte von Spalte WY sind der Arkuskotangens des Inhalts von Spalte W3. Alle Werte sind in Grad angegeben.
TEMP = ACOT(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils den Arkuskotangens der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = ACOT(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der Arkuskotangens der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind der Arkuskotangens der entsprechenden Zeilen von Spalte W2. Alle Werte sind in Radiant angegeben.

Zugehörige Funktionen

Funktion	Beschreibung
ACOS	Berechnet den Arkuskosinus des Inhalts des angegebenen Datenbereichs.
ASIN	Berechnet den Arkussinus des Inhalts des angegebenen Datenbereichs.
ATAN	Berechnet den Arkustangens des Inhalts des angegebenen Datenbereichs.
COT	Berechnet den Kotangens des Inhalts des angegebenen Datenbereichs.

ADD_MONTHS

Syntax

ADD_MONTHS(Monate, Datumszeichenfolge [, Eingabeformat])

Parameter

Monate

Eine Ganzzahl, die eine Anzahl von Monaten darstellt, die zu [Datumszeichenfolge] addiert werden soll.

Datumszeichenfolge

Eine Zeichenfolge, die ein gültiges Datum in dem Format BEGRENZ_M_T_J oder in dem durch das optionale Argument [Eingabeformat] angegebenen Format darstellt.

Eingabeformat

Das Format, das für das berechnete Datum verwendet werden soll. Eine Liste der unterstützten Datumsformate finden Sie unter der Funktion DATE_FORMAT. [Eingabeformat] legt nicht nur das Format der Eingabezeichenfolge, sondern auch das Format der Ausgabezeichenfolge fest.

Beschreibung

ADD_MONTHS gibt ein Datum zurück, das sich aus der Addition der angegebenen Anzahl von Monaten zu der angegebenen Datumszeichenfolge ergibt. Das Datum wird in dem Standardformat BEGRENZ_M_T_J oder in dem durch das optionale Argument [Eingabeformat] angegebenen Format zurückgegeben. Mithilfe von DATE_FORMAT können Sie ein anderes Ausgabeformat festlegen.

Wenn die Erhöhung des Monats um die angegebene Anzahl von Monaten ein ungültiges Datum erzeugt, wird als Ergebnis der letzte Tag des Monats berechnet, wie im letzten der folgenden Beispiele zu sehen. Gegebenenfalls werden auch Schaltjahre berücksichtigt. Wenn beispielsweise zum 31. Januar 2012 ein Monat addiert wird, ergibt sich der 29. Februar 2012.

Beispiele

ADD_MONTHS(12, '06-25-11') addiert zu dem angegebenen Datum ein Jahr (12 Monate) und gibt das Datum 06-25-12 zurück.

ADD_MONTHS(3, '2011-06-25', DT_BEGRENZ_J_M_T) addiert zu dem angegebenen Datum drei Monate und gibt das Datum 2011-09-25 zurück.

ADD_MONTHS(1, '02-28-2011') gibt das Datum 03-28-2011 zurück.

ADD_MONTHS(1, '03-31-2012') gibt das Datum 04-30-2012 zurück.

Zugehörige Funktionen

Funktion	Beschreibung
DATE	Wandelt eine Datumszeichenfolge in ein julianisches Datum um.
DATE_FORMAT	Wandelt ein Datum von Eingabeformat in Ausgabeformat um.

ALIGN



Syntax

ALIGN(Ref_Reihe, Reihe, Bereich)

Parameter

Ref_Reihe

Reihe

Bereich

Beschreibung

Richtet einen Bereich mit der durch seine Reihe angegebenen Position an der Referenzreihe aus; dabei hat jede Zeile in <Bereich> eine entsprechende Nummer, die in <Reihe> angegeben ist. Die Zeile wird am richtigen Zeilenoffset ausgerichtet, indem <Reihe> mit dem Offset in <Ref_Reihe> abgeglichen wird. Wenn <Reihe> nicht die Nummer in <Ref_Reihe> enthält, wird 0 eingefügt. Wenn <Ref_Reihe> nicht die Nummer in <Reihe> enthält, wird die Zeile ignoriert. Wenn mehrere Zeilen an der Referenzreihe ausgerichtet werden können, wird die erste Zeile verwendet. Mit dieser Funktion können Datumsangaben, die durch Zahlen ausgedrückt sind, einer Zeitreihe zugeordnet werden, die bei einem anderen Datum beginnt. Auf diese Weise kann ein Block generiert werden, in dem jede Zeile Daten aus demselben Tag enthält.

AND

Syntax

```
Daten1 AND Daten2; Daten1 && Daten2
```

Parameter

Daten1

Die Zahlen, die durch logisches Und mit den Werten in Daten2 verknüpft werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makro-funktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Die Zahl(en), die durch logisches Und mit den Werten in Daten1 verknüpft werden soll(en). Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Daten2 muss mit der Anzahl der Spalten in Daten1 übereinstimmen, es sei denn, bei Daten2 handelt es sich um eine Konstante. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makro-funktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

AND berechnet das logische Und zwischen den beiden angegebenen Datenbereichen. AND gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten1 durch logisches Und mit der entsprechenden Spalte von Daten2 verknüpft wird (d. h., die erste Spalte von Daten1 wird durch logisches Und mit der ersten Spalte von Daten2 verknüpft, die zweite Spalte mit der zweiten Spalte usw.).

Wenn es sich bei Daten2 um eine Konstante handelt, wird jeder Wert in Daten1 durch logisches Und mit dem betreffenden Wert verknüpft. Wenn Daten2 eine oder mehrere Spalten enthält, werden die Berechnungen zeilenweise zwischen einer Spalte von Daten1 und einer Spalte von Daten2 ausgeführt. Die erste Zeile von Daten1 wird durch logisches Und mit der ersten Zeile von Daten2 verknüpft, die zweite Zeile mit der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Der Operator AND kann durch ein doppeltes Et-Zeichen (&&) abgekürzt werden. Mit dem doppelten Et-Zeichen können die beiden Argumente getrennt werden (statt z. B. W1 AND 3 anzugeben, können Sie einfach W1&&3 eingeben).

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = 1 AND 8 oder TEMP = 1 && 8
Erstellt eine neue Spalte TEMP, die den Wert eins enthält (jede Zahl ungleich null wird als eins behandelt).
TEMP = W1 && 1
Erstellt eine neue Spalte TEMP, die für jeden Wert von Spalte W1 den Wert eins enthält.
TEMP = W1 && W1
Erstellt eine neue Spalte TEMP, die für jeden Wert ungleich null in Spalte W1 den Wert eins und für jede Null in Spalte W1 den Wert null enthält.
TEMP = W1 && W2
Erstellt eine neue Spalte TEMP, in der jeder Zeilenwert von Spalte W1 durch logisches Und mit dem entsprechenden Zeilenwert von Spalte W2 verknüpft ist.
TEMP = W1:W3 && W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Werte von Spalte W1 durch logisches Und mit den entsprechenden Zeilenwerten von Spalte W4 verknüpft. In der Spalte WX sind die Werte von Spalte W2 durch logisches Und mit den Werten von Spalte W5 verknüpft. In der Spalte WY sind die Werte von Spalte W3 durch logisches Und mit den Werten von Spalte W6 verknüpft.
TEMP = W1[10:20] && W2 oder TEMP = W1[10:20] && W2[1:11]
Erstellt eine neue Spalte TEMP, in deren ersten 11 Zellen die Werte in Zeile 10-20 von Spalte W1 durch logisches Und mit den Werten in Zeile 1-11 von Spalte W2 verknüpft sind. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
NOT	Berechnet das logische Nicht des Inhalts des angegebenen Datenbereichs.
OR	Berechnet das logische Oder zwischen den beiden angegebenen Datenbereichen.

ASIN

Syntax

ASIN(Daten [, Einheit])

Parameter

Daten

Die numerischen Werte, deren Arkussinus berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Einheit

Dieses optionale Schlüsselwort legt fest, ob die Eingabewerte und Ergebnisse als Grad oder als Radiant interpretiert werden. Wählen Sie eine der folgenden Angaben aus:

RADIAN - Führt die Berechnungen in Radiant aus (Standard)

DEGREE - Führt die Berechnungen in Grad aus

Wird dieser Parameter nicht angegeben, ist der Standard Radiant. (Die Umrechnung von Radiant in Grad erfolgt durch Division durch π und Multiplikation mit 180.)

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

ASIN berechnet den Arkussinus der Werte im angegebenen Datenbereich. Der Arkussinus ist der Winkel, dessen Sinus der Inhalt der jeweiligen Zelle ist. ASIN gibt für jede Eingabespalte eine neue Spalte zurück, die den Arkussinus der Zahlen in der entsprechenden Eingabespalte enthält.

Wenn das Schlüsselwort RADIAN verwendet wird, gibt ASIN Werte im Bereich $-\pi/2$ bis $\pi/2$ zurück. Wenn das Schlüsselwort DEGREE verwendet wird, gibt ASIN Werte im Bereich -90 bis 90 zurück.

Anmerkung: Der Zelleninhalt jeder angegebenen Spalte muss Werte zwischen -1,0 und 1,0 (einschließlich) aufweisen. Andernfalls wird für jede ungültige Eingabe ??? zurückgegeben.

Beispiele

TEMP = ASIN(0,5) oder TEMP = ASIN(0,5, 0) oder TEMP = ASIN(0,5, RADIAN)
Erstellt eine neue Spalte TEMP, die den Wert 0,524 ($\pi/6$ Radiant) enthält.
TEMP = ASIN(0,5, 1) oder TEMP = ASIN(0,5, DEGREE)
Erstellt eine neue Spalte TEMP, die den Wert 30 (Grad) enthält.
TEMP = ASIN(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert den Arkussinus (in Radiant) des Inhalts von Spalte W1 darstellt.

TEMP = ASIN(W1:W3, 1)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der Arkussinus des Inhalts von Spalte W1, die Werte von Spalte WX sind der Arkussinus des Inhalts von Spalte W2 und die Werte von Spalte WY sind der Arkussinus des Inhalts von Spalte W3. Alle Werte sind in Grad angegeben.
TEMP = ASIN(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils den Arkussinus der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = ASIN(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der Arkussinus der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind der Arkussinus der entsprechenden Zeilen von Spalte W2. Alle Werte sind in Radiant angegeben.

Zugehörige Funktionen

Funktion	Beschreibung
ACOS	Berechnet den Arkuskosinus des Inhalts des angegebenen Datenbereichs.
ACOT	Berechnet den Arkuskotangens des Inhalts des angegebenen Datenbereichs.
ATAN	Berechnet den Arkustangens des Inhalts des angegebenen Datenbereichs.
SIN	Berechnet den Sinus des Inhalts des angegebenen Datenbereichs.

ATAN

Syntax

ATAN(Daten [, Einheit])

Parameter

Daten

Die numerischen Werte, deren Arkustangens berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Einheit

Dieses optionale Schlüsselwort legt fest, ob die Eingabewerte und Ergebnisse als Grad oder als Radiant interpretiert werden. Wählen Sie eine der folgenden Angaben aus:

RADIAN - Führt die Berechnungen in Radiant aus (Standard)

DEGREE - Führt die Berechnungen in Grad aus

Wird dieser Parameter nicht angegeben, ist der Standard Radiant. (Die Umrechnung von Radiant in Grad erfolgt durch Division durch Pi und Multiplikation mit 180.)

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

ATAN berechnet den Arkustangens der Werte im angegebenen Datenbereich. Der Arkustangens ist der Winkel, dessen Tangens der Inhalt der jeweiligen Zelle ist. ATAN gibt für jede Eingabespalte eine neue Spalte zurück, die den Arkustangens der Zahlen in der entsprechenden Eingabespalte enthält.

Wenn das Schlüsselwort RADIAN verwendet wird, gibt ATAN Werte im Bereich $-\pi/2$ bis $\pi/2$ zurück. Wenn das Schlüsselwort DEGREE verwendet wird, gibt ATAN Werte im Bereich -90 bis 90 zurück.

Beispiele

TEMP = ATAN(1) oder TEMP = ATAN(1, 0) oder TEMP = ATAN(1, RADIAN)
Erstellt eine neue Spalte TEMP, die den Wert 0,785 ($\pi/4$ Radiant) enthält.
TEMP = ATAN(1, 1) oder TEMP = ATAN(1, DEGREE)
Erstellt eine neue Spalte TEMP, die den Wert 45 (Grad) enthält.
TEMP = ATAN(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert den Arkustangens (in Radiant) des Inhalts von Spalte W1 darstellt.
TEMP = ATAN(W1:W3, 1)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der Arkustangens des Inhalts von Spalte W1, die Werte von Spalte WX sind der Arkustangens des Inhalts von Spalte W2 und die Werte von Spalte WY sind der Arkustangens des Inhalts von Spalte W3. Alle Werte sind in Grad angegeben.
TEMP = ATAN(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils den Arkustangens der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = ATAN(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der Arkustangens der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind der Arkustangens der entsprechenden Zeilen von Spalte W2. Alle Werte sind in Radiant angegeben.

Zugehörige Funktionen

Funktion	Beschreibung
ACOS	Berechnet den Arkuskosinus des Inhalts des angegebenen Datenbereichs.

Funktion	Beschreibung
ASIN	Berechnet den Arkussinus des Inhalts des angegebenen Datenbereichs.
ATAN	Berechnet den Arkustangens des Inhalts des angegebenen Datenbereichs.
TAN	Berechnet den Tangens des Inhalts des angegebenen Datenbereichs.

AVG

Syntax

AVG(Daten [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, deren arithmetisches Mittel berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabedatenbereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Anmerkung: Viele Makrofunktionen setzen die Schlüsselwortparameter {ALL | COL | ROW} ein. Diese Schlüsselwörter gelten nicht für **IBM Kampagne**, da es sich bei den Eingabedaten immer um eine Einzelspalte oder ein Einzelfeld handelt. Das Makro verhält sich immer so, als ob das Schlüsselwort COL angegeben würde. Deshalb brauchen Sie diese Schlüsselwörter nicht anzugeben, wenn Sie **IBM Kampagne** verwenden.

Beschreibung

AVG berechnet das arithmetische Mittel oder den Durchschnitt der Zellen im angegebenen Datenbereich. Das arithmetische Mittel wird berechnet, indem der Inhalt aller Zellen addiert und durch die Anzahl der Zellen geteilt wird. Die Anzahl der von AVG zurückgegebenen Spalten hängt von Schlüsselwort ab.

- Wenn Schlüsselwort den Wert ALL hat, gibt AVG eine neue Spalte zurück, die einen Einzelwert (den Durchschnitt aller Zellen in Daten) enthält.
- Wenn Schlüsselwort den Wert COL hat, gibt AVG_DEV für jede Eingabespalte eine neue Spalte zurück. Jede neue Spalte enthält einen Wert (den Durchschnitt aller Zellen in der entsprechenden Eingabespalte).
- Wenn Schlüsselwort den Wert ROW hat, gibt AVG eine neue Spalte zurück, die den Durchschnitt für jede Zeile von Daten enthält.

Anmerkung: Leere Zellen werden bei der Berechnung ignoriert.

Anmerkung: AVG ist mit der Makrofunktion MEAN identisch.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = AVG(MERGE(3, 4, 5)) oder TEMP = AVG(MERGE(3, 4, 5), ALL)
Erstellt eine neue Spalte TEMP, die den Wert 4 enthält.
TEMP = AVG(MERGE(-10, 3, 10))Erstellt eine neue Spalte TEMP, die den Wert 1 enthält.
TEMP = AVG(W1)Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das arithmetische Mittel des Inhalts von Spalte W1 darstellt.
TEMP = AVG(W1:W3)Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das arithmetische Mittel des Inhalts der Spalten W1, W2 und W3 darstellt.
TEMP = AVG(W1[10:20])Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das arithmetische Mittel der Zellen in Zeile 10-20 von Spalte W1 darstellt.
TEMP = AVG(W1[1:5]:W4)Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das arithmetische Mittel der Zellen in Zeile 1-5 von Spalte W1 bis W4 darstellt.
TEMP = AVG(W1:W3, COL)Erstellt drei neue Spalten TEMP, WX und WY. Der Einzelwert in der Spalte TEMP ist das arithmetische Mittel des Inhalts von Spalte W1, der Einzelwert in der Spalte WX ist das arithmetische Mittel des Inhalts von Spalte W2 und der Einzelwert in der Spalte WY ist das arithmetische Mittel des Inhalts von Spalte W3.
TEMP = AVG(MERGE(1,4), COL)Erstellt zwei neue Spalten TEMP und WX. TEMP enthält den Einzelwert eins; WX enthält den Einzelwert vier.
TEMP = AVG(W1[1:5]:W3, COL)Erstellt drei neue Spalten TEMP, WX und WY, die jeweils einen Einzelwert enthalten. Der Wert in der Spalte TEMP ist das arithmetische Mittel der Zellen in Zeile 1-5 von Spalte W1, der Wert in der Spalte WX ist das arithmetische Mittel der Zellen in Zeile 1-5 von Spalte W2 und der Wert in der Spalte WY ist das arithmetische Mittel der Zellen in Zeile 1-5 von Spalte W3.
TEMP = AVG(W1, ROW)Erstellt eine neue Spalte TEMP, die dieselben Werte wie Spalte W1 enthält (das arithmetische Mittel einer beliebigen Zahl ist die Zahl selbst).
TEMP = AVG(W1:W3, ROW)Erstellt eine neue Spalte TEMP, in der jeder Zelleintrag das arithmetische Mittel der entsprechenden Zeile von Spalte W1, W2 und W3 darstellt.
TEMP = AVG(W1[1:5]:W3, ROW)Erstellt eine neue Spalte TEMP, in der die Zellen in Zeile 1-5 das arithmetische Mittel der entsprechenden Zeile von Spalte W1 bis W3 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
AVG_DEV	Berechnet die mittlere Abweichung eines Zellenbereichs.

Funktion	Beschreibung
SUM oder TOTAL	Berechnet die Summe eines Zellenbereichs.

AVG_DEV



Syntax

AVG_DEV(Daten [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, deren mittlere Abweichung berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabedatenbereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

AVG_DEV berechnet die mittlere Abweichung der Werte im angegebenen Datenbereich vom Durchschnitt. Das arithmetische Mittel wird berechnet, indem der Inhalt aller Zellen addiert und durch die Anzahl der Zellen geteilt wird. Die vollständige Gleichung für AVG_DEV lautet wie folgt:

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

Die Anzahl der von AVG zurückgegebenen Spalten hängt von Schlüsselwort ab.

- Wenn Schlüsselwort den Wert ALL hat, gibt AVG_DEV eine neue Spalte zurück, die einen Einzelwert (die mittlere Abweichung aller Zellen in Daten) enthält.
- Wenn Schlüsselwort den Wert COL hat, gibt AVG_DEV für jede Eingabespalte eine neue Spalte zurück. Jede neue Spalte enthält einen Wert (die mittlere Abweichung aller Zellen in der entsprechenden Eingabespalte).
- Wenn Schlüsselwort den Wert ROW hat, gibt AVG_DEV eine neue Spalte zurück, die die mittlere Abweichung für jede Zeile von Daten enthält.

Anmerkung: Leere Zellen werden bei der Berechnung des Durchschnitts ignoriert.

Beispiele

TEMP = AVG_DEV(MERGE(3, 4, 5)) oder TEMP = AVG_DEV(MERGE(3, 4, 5), ALL)
Erstellt eine neue Spalte TEMP, die den Wert 0,67 enthält.
TEMP = AVG_DEV(COLUMN(-4, 0))
Erstellt eine neue Spalte TEMP, die den Wert 2 enthält.
TEMP = AVG_DEV(W1)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die mittlere Abweichung des Inhalts von Spalte W1 darstellt.
TEMP = AVG_DEV(W1:W3)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die mittlere Abweichung des Inhalts der Spalten W1, W2 und W3 darstellt.
TEMP = AVG_DEV(W1[10:20])
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die mittlere Abweichung der Zellen in Zeile 10-20 von Spalte W1 darstellt.
TEMP = AVG_DEV(W1[1:5]:W4)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die mittlere Abweichung der Zellen in Zeile 1-5 von Spalte W1 bis W4 darstellt.
TEMP = AVG_DEV(W1:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY. Der Einzelwert in der Spalte TEMP ist die mittlere Abweichung des Inhalts von Spalte W1, der Einzelwert in der Spalte WX ist die mittlere Abweichung des Inhalts von Spalte W2 und der Einzelwert in der Spalte WY ist die mittlere Abweichung des Inhalts von Spalte W3.
TEMP = AVG_DEV(MERGE(1,4), COL)
Erstellt zwei neue Spalten TEMP und WX, die beide den Einzelwert 0 enthalten.
TEMP = AVG_DEV(W1[1:5]:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils einen Einzelwert enthalten. Der Wert in der Spalte TEMP ist die mittlere Abweichung der Zellen in Zeile 1-5 von Spalte W1, der Wert in der Spalte WX ist die mittlere Abweichung der Zellen in Zeile 1-5 von Spalte W2 und der Wert in der Spalte WY ist die mittlere Abweichung der Zellen in Zeile 1-5 von Spalte W3.
TEMP = AVG_DEV(W1, ROW)
Erstellt eine neue Spalte TEMP, die für jeden Wert von Spalte W1 eine Null enthält (die mittlere Abweichung einer beliebigen Zahl ist null).
TEMP = AVG_DEV(W1:W3, ROW)
Erstellt eine neue Spalte TEMP, in der jeder Zelleneintrag die mittlere Abweichung der entsprechenden Zeile in Spalte W1, W2 und W3 darstellt.

```
TEMP = AVG_DEV(W1[1:5]:W3, ROW)
```

Erstellt eine neue Spalte TEMP, in der die Zellen in Zeile 1-5 die mittlere Abweichung der entsprechenden Zeile in Spalte W1 bis W3 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
AVG oder MEAN	Berechnet das arithmetische Mittel oder den Durchschnitt eines Zellenbereichs.
SUM oder TOTAL	Berechnet die Summe eines Zellenbereichs.

BETWEEN

Syntax

```
Wert1 BETWEEN Wert2 AND Wert3
```

Parameter

Äquivalent zu Wert1 >= Wert2 AND < Wert3

Beschreibung

BETWEEN ist eine spezielle Variante des Vergleichsprädikats. Es kommt auf die Details dieses Prädikats an; die Reihenfolge der Operanden hat einige unerwartete Auswirkungen. Vergleichen Sie dazu die nachfolgenden Beispiele.

Anmerkung: FROM und FOR verwenden eine identische Syntax.

Beispiele

```
SELECT * FROM Filmtitel WHERE unsere_Kosten BETWEEN 11,00 AND 27,50 ;
```

gibt eine Liste der Filme zurück, die nicht weniger als 11,00 und nicht mehr als 27,50 kosten.

```
10 BETWEEN 5 AND 15 Ist wahr, aber: 10 BETWEEN 15 AND 5 Ist falsch:
```

da der entsprechende Ausdruck mit BETWEEN und AND eine bestimmte Reihenfolge aufweist, die keine Rolle spielt, wenn Sie Literale verwenden, aber von großer Bedeutung ist, wenn Sie durch Variablen, Parameter oder sogar Unterabfragen Wert2 und Wert3 angeben.

BIT_AND

Syntax

```
Daten1 BIT_AND Daten2; Daten1 & Daten2
```

Parameter

Daten1

Die nicht negativen Ganzzahlen, die durch bitweises Und mit den Werten in Daten2 verknüpft werden sollen. Dabei kann es sich um einen konstanten Wert,

eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Die nicht negative(n) Ganzzahl(en), die durch bitweises Und mit den Werten in Daten1 verknüpft werden soll(en). Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Daten2 muss mit der Anzahl der Spalten in Daten1 übereinstimmen, es sei denn, bei Daten2 handelt es sich um eine Konstante. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

BIT_AND berechnet das bitweise Und zwischen den beiden angegebenen Datenbereichen. BIT_AND gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten1 durch bitweises Und mit der entsprechenden Spalte von Daten2 verknüpft wird (d. h., die erste Spalte von Daten1 wird durch bitweises Und mit der ersten Spalte von Daten2 verknüpft, die zweite Spalte mit der zweiten Spalte usw.).

Wenn es sich bei Daten2 um eine Konstante handelt, wird jeder Wert in Daten1 durch bitweises Und mit dem betreffenden Wert verknüpft. Wenn Daten2 eine oder mehrere Spalten enthält, werden die Berechnungen zeilenweise zwischen einer Spalte von Daten1 und einer Spalte von Daten2 ausgeführt. Die erste Zeile von Daten1 wird durch bitweises Und mit der ersten Zeile von Daten2 verknüpft, die zweite Zeile mit der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Bei dieser Makrofunktion ist die Genauigkeit auf Ganzzahlwerte kleiner als 2^{24} begrenzt. Negative Werte sind nicht zulässig.

Anmerkung: Der Operator BIT_AND kann durch ein Et-Zeichen (&) abgekürzt werden. Mit dem Et-Zeichen können die beiden Argumente getrennt werden (statt z. B. BIT_AND(W1, 3) anzugeben, können Sie einfach W1&3 eingeben).

Beispiele

```
TEMP = 3 BIT_AND 7 oder TEMP = 3 & 7
```

Erstellt eine neue Spalte TEMP, die den Wert drei enthält (das bitweise Und von 011 und 111 ergibt 011).

```
TEMP = W1 & 8
```

Erstellt eine neue Spalte TEMP, in der der Inhalt von Spalte W1 jeweils durch bitweises Und mit dem Binärwert 1000 verknüpft ist.

```
TEMP = W1 & W1
```

Erstellt eine neue Spalte TEMP, die denselben Inhalt wie die Spalte W1 enthält (jeder Wert ergibt bei Verknüpfung durch AND mit sich selbst den Wert selbst).

TEMP = W1 & W2
Erstellt eine neue Spalte TEMP, in der jeder Zeilenwert von Spalte W1 durch bitweises Und mit dem entsprechenden Zeilenwert von Spalte W2 verknüpft ist.
TEMP = W1:W3 & W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Werte von Spalte W1 durch bitweises Und mit den entsprechenden Zeilenwerten von Spalte W4 verknüpft. In der Spalte WX sind die Werte von Spalte W2 durch bitweises Und mit den Werten von Spalte W5 verknüpft. In der Spalte WY sind die Werte von Spalte W3 durch bitweises Und mit den Werten von Spalte W6 verknüpft.
TEMP = W1[10:20] & W2 oder TEMP = W1[10:20] & W2[1:11]
Erstellt eine neue Spalte TEMP, in deren ersten 11 Zellen die Werte in Zeile 10-20 von Spalte W1 durch bitweises Und mit den Werten in Zeile 1-11 von Spalte W2 verknüpft sind. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
BIT_NOT	Berechnet das bitweise Nicht des Inhalts des angegebenen Datenbereichs.
BIT_OR	Berechnet das bitweise Oder zwischen den beiden angegebenen Datenbereichen.
BIT_XOR oder XOR	Berechnet das bitweise exklusive Oder zwischen den beiden angegebenen Datenbereichen.

BIT_NOT

Syntax

BIT_NOT Daten; ~ Daten

Parameter

Daten

Die nicht negativen Ganzzahlen, die durch bitweises Nicht verknüpft werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

BIT_NOT gibt das bitweise Nicht der Werte im angegebenen Datenbereich zurück. BIT_NOT gibt für jede Eingabespalte eine neue Spalte zurück, die das bitweise Nicht der Werte in der entsprechenden Spalte von Daten enthält.

Anmerkung: Bei dieser Makrofunktion ist die Genauigkeit auf Ganzzahlwerte kleiner als 2^{24} begrenzt. Negative Werte sind nicht zulässig.

Anmerkung: Wenn eine Spalte in jeder Zeile dieselbe Zahl x wie Daten enthält, ist dies dasselbe, als wenn als Daten die Konstante x verwendet wird.

Anmerkung: Der Operator BIT_NOT kann durch eine Tilde (~) abgekürzt werden. Stellen Sie die Tilde dem Datenwert voran (statt z. B. BIT_NOT(W1) anzugeben, können Sie einfach ~W1 eingeben).

Beispiele

TEMP = BIT_NOT 3 oder TEMP = ~3
Erstellt eine neue Spalte TEMP, die den Wert vier enthält (das bitweise Nicht von 011 ergibt 100).
TEMP = ~W1
Erstellt eine neue Spalte TEMP, in der jeder Wert das bitweise Nicht des Inhalts von Spalte W1 darstellt.
TEMP = ~W1:W3
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils das bitweise Nicht des Inhalts von Spalte W1, die Werte von Spalte WX sind das bitweise Nicht des Inhalts von Spalte W2 und die Werte von Spalte WY sind das bitweise Nicht des Inhalts von Spalte W3.
TEMP = ~W1[100:200]
Erstellt eine neue Spalte TEMP, in der die ersten 101 Zellen jeweils das bitweise Nicht der Werte in Zeile 100-200 von Spalte W1 darstellen.

Zugehörige Funktionen

Funktion	Beschreibung
BIT_AND	Berechnet das bitweise Und zwischen den beiden angegebenen Datenbereichen.
BIT_OR	Berechnet das bitweise Oder zwischen den beiden angegebenen Datenbereichen.
BIT_XOR oder XOR	Berechnet das bitweise exklusive Oder zwischen den beiden angegebenen Datenbereichen.

BIT_OR

Syntax

Daten1 BIT_OR Daten2; Daten1 OR Daten2; Daten1 | Daten2

Parameter

Daten1

Die nicht negativen Ganzzahlen, die durch bitweises Oder mit den Werten in Daten2 verknüpft werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Die nicht negative(n) Ganzzahl(en), die durch bitweises Oder mit den Werten in Daten1 verknüpft werden soll(en). Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Daten2 muss mit der Anzahl der Spalten in Daten1 übereinstimmen, es sei denn, bei Daten2 handelt es sich um eine Konstante. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

BIT_OR berechnet das bitweise Oder zwischen den beiden angegebenen Datenbereichen. BIT_OR gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten1 durch bitweises Oder mit der entsprechenden Spalte von Daten2 verknüpft wird (d. h., die erste Spalte von Daten1 wird durch bitweises Oder mit der ersten Spalte von Daten2 verknüpft, die zweite Spalte mit der zweiten Spalte usw.).

Wenn es sich bei Daten2 um eine Konstante handelt, wird jeder Wert in Daten1 durch bitweises Oder mit dem betreffenden Wert verknüpft. Wenn Daten2 eine oder mehrere Spalten enthält, werden die Berechnungen zeilenweise zwischen einer Spalte von Daten1 und einer Spalte von Daten2 ausgeführt. Die erste Zeile von Daten1 wird durch bitweises Oder mit der ersten Zeile von Daten2 verknüpft, die zweite Zeile mit der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Bei dieser Makrofunktion ist die Genauigkeit auf Ganzzahlwerte kleiner als 2^{24} begrenzt. Negative Werte sind nicht zulässig.

Anmerkung: Der Operator BIT_OR kann durch einen vertikalen Strich (|) abgekürzt werden. Mit dem vertikalen Strich können die beiden Argumente getrennt werden (statt z. B. BIT_OR(W1, 3) anzugeben, können Sie einfach W1|3 eingeben). Sie können auch OR verwenden.

Beispiele

TEMP = 3 BIT_OR 7 oder TEMP = 3 OR 7 oder TEMP = 3 7
Erstellt eine neue Spalte TEMP, die den Wert sieben enthält (das bitweise Oder von 011 und 111 ergibt 111).
TEMP = W1 8
Erstellt eine neue Spalte TEMP, in der der Inhalt von Spalte W1 jeweils durch bitweises Oder mit dem Binärwert 1000 verknüpft ist.
TEMP = W1 W1
Erstellt eine neue Spalte TEMP, die denselben Inhalt wie die Spalte W1 enthält (jeder Wert ergibt bei Verknüpfung durch OR mit sich selbst den Wert selbst).
TEMP = W1 W2
Erstellt eine neue Spalte TEMP, in der jeder Zeilenwert von Spalte W1 durch bitweises Oder mit dem entsprechenden Zeilenwert von Spalte W2 verknüpft ist.

TEMP = W1:W3 | W4:W6

Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Werte von Spalte W1 durch logisches Oder mit den entsprechenden Zeilenwerten von Spalte W4 verknüpft. In der Spalte WX sind die Werte von Spalte W2 durch logisches Oder mit den Werten von Spalte W5 verknüpft. In der Spalte WY sind die Werte von Spalte W3 durch logisches Oder mit den Werten von Spalte W6 verknüpft.

TEMP = W1[10:20] | W2 oder TEMP = W1[10:20] | W2[1:11]

Erstellt eine neue Spalte TEMP, in deren ersten 11 Zellen die Werte in Zeile 10-20 von Spalte W1 durch bitweises Oder mit den Werten in Zeile 1-11 von Spalte W2 verknüpft sind. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
BIT_AND	Berechnet das bitweise Und zwischen den beiden angegebenen Datenbereichen.
BIT_NOT	Berechnet das bitweise Nicht des Inhalts des angegebenen Datenbereichs.
BIT_XOR oder XOR	Berechnet das bitweise exklusive Oder zwischen den beiden angegebenen Datenbereichen.

BIT_XOR

Syntax

Daten1 BIT_XOR Daten2

Parameter

Daten1

Die nicht negativen Ganzzahlen, die durch bitweises exklusives Oder mit den Werten in Daten2 verknüpft werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Die nicht negative(n) Ganzzahl(en), die durch bitweises exklusives Oder mit den Werten in Daten1 verknüpft werden soll(en). Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Daten2 muss mit der Anzahl der Spalten in Daten1 übereinstimmen, es sei denn, bei Daten2 handelt es sich um eine Konstante. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

BIT_XOR berechnet das bitweise exklusive Oder zwischen den beiden angegebenen Datenbereichen. BIT_XOR gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten1 durch bitweises exklusives Oder mit der entsprechenden Spalte von Daten2 verknüpft wird (d. h., die erste Spalte von Daten1 wird durch bitweises exklusives Oder mit der ersten Spalte von Daten2 verknüpft, die zweite Spalte mit der zweiten Spalte usw.).

Wenn es sich bei Daten2 um eine Konstante handelt, wird jeder Wert in Daten1 durch bitweises exklusives Oder mit dem betreffenden Wert verknüpft. Wenn Daten2 eine oder mehrere Spalten enthält, werden die Berechnungen zeilenweise zwischen einer Spalte von Daten1 und einer Spalte von Daten2 ausgeführt. Die erste Zeile von Daten1 wird durch bitweises exklusives Oder mit der ersten Zeile von Daten2 verknüpft, die zweite Zeile mit der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Bei dieser Makrofunktion ist die Genauigkeit auf Ganzzahlwerte kleiner als 2^{24} begrenzt. Negative Werte sind nicht zulässig.

Beispiele

TEMP = 3 BIT_XOR 7
Erstellt eine neue Spalte TEMP, die den Wert vier enthält (das bitweise exklusive Oder von 011 und 111 ergibt 100).
TEMP = W1 BIT_XOR 8
Erstellt eine neue Spalte TEMP, in der der Inhalt von Spalte W1 jeweils durch bitweises exklusives Oder mit dem Binärwert 1000 verknüpft ist.
TEMP = W1 BIT_XOR W1
Erstellt eine neue Spalte TEMP, die nur Nullen enthält (jeder Wert, der durch exklusives Oder mit sich selbst verknüpft wird, ergibt null).
TEMP = W1 BIT_XOR W2
Erstellt eine neue Spalte TEMP, in der jeder Zeilenwert von Spalte W1 durch bitweises exklusives Oder mit dem entsprechenden Zeilenwert von Spalte W2 verknüpft ist.
TEMP = W1:W3 BIT_XOR W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Werte von Spalte W1 durch bitweises exklusives Oder mit den entsprechenden Zeilenwerten von Spalte W4 verknüpft. In der Spalte WX sind die Werte von Spalte W2 durch bitweises exklusives Oder mit den Werten von Spalte W5 verknüpft. In der Spalte WY sind die Werte von Spalte W3 durch bitweises exklusives Oder mit den Werten von Spalte W6 verknüpft.
TEMP = W1[10:20] BIT_XOR W2 oder TEMP = W1[10:20] BIT_XOR W2[1:11]
Erstellt eine neue Spalte TEMP, in deren ersten 11 Zellen die Werte in Zeile 10-20 von Spalte W1 durch bitweises exklusives Oder mit den Werten in Zeile 1-11 von Spalte W2 verknüpft sind. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
BIT_AND	Berechnet das bitweise Und zwischen den beiden angegebenen Datenbereichen.
BIT_NOT	Berechnet das bitweise Nicht des Inhalts des angegebenen Datenbereichs.
BIT_OR	Berechnet das bitweise Oder zwischen den beiden angegebenen Datenbereichen.

BUFFER



Syntax

BUFFER(Daten)

Parameter

Daten

Die Werte, die als Konstanten kopiert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

BUFFER legt eine dynamische Kopie der Werte im angegebenen Datenbereich an. Diese kopierten Werte ändern sich, wenn sich die entsprechenden Werte der Eingabespalte(n) ändern. BUFFER gibt für jede Eingabespalte eine neue Spalte zurück, die eine Kopie der Werte in der entsprechenden Eingabespalte enthält.

Anmerkung: Die Anwendung der Makrofunktion BUFFER kann die Leistung bei der Ausführung von Experimenten beträchtlich steigern. Wenn die Eingabe- oder Ausgabespalten für ein Experiment auf komplexen (langsamen) Berechnungen im Arbeitsblatt basieren, sollten Sie jeder Spalte die Makrofunktion BUFFER hinzufügen. Diese Funktion berechnet die Werte und speichert die berechneten Werte. Andernfalls müssen die Werte jedes Mal, wenn das Experiment auf Musterdaten des Arbeitsblatts zugreift, neu berechnet werden. Wenn sich einer der Eingabewerte ändert, werden die Daten in BUFFER ebenso wie andere Makrofunktionen im Arbeitsblatt dynamisch aktualisiert.

Anmerkung: Wenn sich die Datenwerte niemals ändern werden, sollten Sie stattdessen die Makrofunktion CONSTANT verwenden. Diese Funktion legt eine statische Kopie des Datenbereichs an.

Anmerkung: Wenn eine Benutzerfunktion auf der Basis einer Funktionsdefinition erstellt wird, die die Makrofunktion BUFFER verwendet, wird der Anteil der Funktionsdefinition, der in der Makrofunktion BUFFER enthalten ist, als Konstante betrachtet. Es sind keine Eingabevariablen erforderlich, um die Benutzerfunktion anzuwenden.

Beispiele

<code>TEMP = BUFFER(4,3)</code>
Erstellt eine neue Spalte TEMP, die den Wert 4,3 enthält.
<code>TEMP = BUFFER(W1)</code>
Erstellt eine neue Spalte TEMP, in der jeder Wert eine Kopie des Inhalts von Spalte W1 darstellt.
<code>TEMP = BUFFER(W1:W3)</code>
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils Kopien des Inhalts von Spalte W1, die Werte von Spalte WX sind Kopien des Inhalts von Spalte W2 und die Werte von Spalte WY sind Kopien des Inhalts von Spalte W3.
<code>TEMP = BUFFER(W1[10:20])</code>
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen Kopien der Werte in Zeile 10-20 von Spalte W1 enthalten. Die anderen Zellen in TEMP sind leer.
<code>TEMP = BUFFER(W1[50:99]:W2)</code>
Erstellt zwei neue Spalten TEMP und WX, die jeweils Werte in Zeile 1-50 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind Kopien der Zeilen von Spalte W1, die Werte in der Spalte WX sind Kopien der Zeilen von Spalte W2.
<code>TEMP = BUFFER(EXTRACT(!ISERROR(W1:W3), W1:W3))</code>
Erstellt drei neue Spalten TEMP, WX und WY. Die drei Spalten enthalten die gültigen Zeilen (d. h. Zeilen, die keine ???-Zellen enthalten) von Spalte W1 bis W3. Die Einfügung der Makrofunktion EXTRACT innerhalb von CONSTANT verhindert, dass die Spalten TEMP, WX und WY neu berechnet werden, wenn sich die Spalten W1 bis W3 ändern, und beugt so der rechenintensiven Ausführung der Makrofunktion EXTRACT vor.

Zugehörige Funktionen

Funktion	Beschreibung
CONSTANT	Kopiert den Eingabebereich einmal (keine dynamische Aktualisierung).

CEILING

Syntax

`CEILING(Daten)`

Parameter

Daten

Die numerischen Werte, deren Obergrenze berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

CEILING berechnet die Obergrenze der Werte im angegebenen Datenbereich. Die Obergrenze einer Zahl ist die kleinste Ganzzahl, die *nicht* kleiner als die Zahl ist. CEILING gibt für jede Eingabespalte eine neue Spalte zurück, die die Obergrenze der Zahlen in der entsprechenden Eingabespalte enthält.

Beispiele

TEMP = CEILING(4,3)
Erstellt eine neue Spalte TEMP, die den Wert 5 enthält.
TEMP = CEILING(-2,9)
Erstellt eine neue Spalte TEMP, die den Wert -2 enthält.
TEMP = CEILING(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert die Obergrenze des Inhalts von Spalte W1 darstellt.
TEMP = CEILING(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils die Obergrenze des Inhalts von Spalte W1, die Werte von Spalte WX sind die Obergrenze des Inhalts von Spalte W2 und die Werte von Spalte WY sind die Obergrenze des Inhalts von Spalte W3.
TEMP = CEILING(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils die Obergrenze der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = CEILING(W1[50:99]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils Werte in Zeile 1-50 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind jeweils die Obergrenze der Zeilen von Spalte W1, die Werte in der Spalte WX sind die Obergrenze der Zeilen von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
FLOOR oder INT	Berechnet die Untergrenze jedes Werts im angegebenen Datenbereich.
FRACTION	Berechnet die Nachkommastellen jedes Werts im angegebenen Datenbereich.
TRUNCATE	Berechnet den Ganzzahlanteil jedes Werts im angegebenen Datenbereich.

COLUMN

Syntax

COLUMN(Daten [, Daten]...) oder (Daten [, Daten]...)

Parameter

Daten

Ein Wert, der beim Erstellen einer Spalte verwendet werden soll. Dabei kann es sich um einen konstanten Wert (numerisch oder ASCII-Text in Anführungszeichen), eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Dieser Parameter kann mehrmals wiederholt werden, doch nachfolgende Parameter müssen dieselbe Dimensionalität (d. h. Spaltenbreite) wie der erste Parameter aufweisen. Alle Werte in allen Daten-Parametern müssen entweder numerisch oder ASCII-Text sein (d. h., Zahlen- und Textwerte dürfen nicht gemischt werden). Wenn mehrere Daten-Parameter angegeben werden, müssen alle dieselbe Anzahl von Spalten aufweisen. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

COLUMN verkettet die Eingaben vertikal zu Spalten einer Funktionsgruppe. Dabei wird dieselbe Anzahl neuer Spalten zurückgegeben, wie im jeweiligen Eingabeparameter angegeben. Es kann eine unbegrenzte Zahl von Argumenten angegeben werden. Alle Argumente müssen entweder numerisch oder ASCII-Zeichenfolgen sein (d. h., Zahlen- und Textwerte dürfen nicht gemischt werden).

Anmerkung: Die Makrofunktion COLUMN kann abgekürzt werden, indem die Daten-Argumente innerhalb runder Klammern durch Kommas getrennt aufgelistet werden (z. B. TEMP = MEAN((1,2,3,4), ALL)). Wenn es nicht um den Einsatz innerhalb einer anderen Makrofunktion geht, sind die Klammern nicht erforderlich; beispielsweise ist W1=1,2,3 äquivalent zu W1=COLUMN(1,2,3).

Beispiele

TEMP = COLUMN(3, 4, 5) oder TEMP = (3,4,5) oder TEMP = 3,4,5
Erstellt eine neue Spalte TEMP, deren erste drei Zellen die Werte 3, 4 und 5 enthalten.
TEMP = COLUMN("eins", "zwei", "drei")
Erstellt eine neue Spalte TEMP, deren erste drei Zellen die Werte eins, zwei und drei enthalten.
TEMP = AVG(W1), STDV(W1)
Erstellt eine neue Spalte TEMP, die in der ersten Zelle den Durchschnitt von Spalte W1 und in der zweiten Zelle die Standardabweichung von Spalte W1 enthält.
TEMP = W1:W2, W3:W4
Erstellt zwei neue Spalten TEMP und WX. Die Spalte TEMP enthält die Werte aus Spalte W1, gefolgt von den Werten aus Spalte W3. Die Spalte WX enthält die Werte aus Spalte W2, gefolgt von den Werten aus Spalte W4.
TEMP = W1:W2, W3:W4
Erstellt zwei neue Spalten TEMP und WX. Die Spalte TEMP enthält die Werte aus Zelle 1-10 von Spalte W1, gefolgt von allen Werten aus Spalte W3. Die Spalte WX enthält die Werte aus Zelle 1-10 von Spalte W2, gefolgt von allen Werten aus Spalte W4.
TEMP = W1:W2, W3:W4
Erstellt zwei neue Spalten TEMP und WX, die jeweils einen Einzelwert enthalten. Die Spalte TEMP enthält den Durchschnitt der Spalten W1 und W2. Die Spalte WX enthält den Durchschnitt der Spalten W3 und W4.

Zugehörige Funktionen

Funktion	Beschreibung
MERGE	Erstellt eine Datengruppe durch horizontale Verkettung der Eingabewerte.
TO	Operator zur Bereichsgenerierung.
TRANSDPOSE	Setzt einen angegebenen Datenbereich um.

CONSTANT



Syntax

CONSTANT (Daten)

Parameter

Daten

Die Werte, die als Konstanten kopiert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

CONSTANT legt eine statische Kopie der Werte im angegebenen Datenbereich an. Diese kopierten Werte ändern sich *nicht*, wenn sich die entsprechenden Werte der Eingabespalte(n) ändern. Die Datenwerte werden einmalig kopiert, wenn die Makrofunktion angewendet wird. CONSTANT gibt für jede Eingabespalte eine neue Spalte zurück, die eine statische Kopie der Werte in der entsprechenden Eingabespalte enthält.

Anmerkung: Die Anwendung der Makrofunktion CONSTANT kann die Leistung bei der Ausführung von Experimenten beträchtlich steigern. Wenn die Eingabe- oder Ausgabespalten für ein Experiment auf komplexen (langsamen) Berechnungen im Arbeitsblatt basieren, sollten Sie jeder Spalte die Makrofunktion CONSTANT hinzufügen. Diese Funktion berechnet einmalig die Werte und speichert die berechneten Werte. Andernfalls müssen die Werte jedes Mal, wenn das Experiment auf Musterdaten des Arbeitsblatts zugreift, neu berechnet werden.

Anmerkung: Wenn sich die Datenwerte ändern können, sollten Sie stattdessen die Makrofunktion BUFFER verwenden. Dadurch wird eine dynamische Kopie des Datenbereichs angelegt, bei der sich die kopierten Werte ändern, wenn sich die entsprechenden Eingabewerte ändern.

Anmerkung: Wenn eine Benutzerfunktion auf der Basis einer Funktionsdefinition erstellt wird, die die Makrofunktion CONSTANT verwendet, wird der Anteil der Funktionsdefinition, der in der Makrofunktion CONSTANT enthalten ist, als Konstante betrachtet. Es sind keine Eingabevariablen erforderlich, um die Benutzerfunktion anzuwenden.

Beispiele

<code>TEMP = CONSTANT(4,3)</code>
Erstellt eine neue Spalte TEMP, die den Wert 4,3 enthält.
<code>TEMP = CONSTANT(W1)</code>
Erstellt eine neue Spalte TEMP, in der jeder Wert eine statische Kopie des Inhalts von Spalte W1 darstellt.
<code>TEMP = CONSTANT(W1:W3)</code>
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils Kopien des Inhalts von Spalte W1, die Werte von Spalte WX sind Kopien des Inhalts von Spalte W2 und die Werte von Spalte WY sind Kopien des Inhalts von Spalte W3.
<code>TEMP = CONSTANT(W1[10:20])</code>
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen Kopien der Werte in Zeile 10-20 von Spalte W1 enthalten. Die anderen Zellen in TEMP sind leer.
<code>TEMP = CONSTANT(W1[50:99]:W2)</code>
Erstellt zwei neue Spalten TEMP und WX, die jeweils Werte in Zeile 1-50 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind Kopien der Zeilen von Spalte W1, die Werte in der Spalte WX sind Kopien der Zeilen von Spalte W2.
<code>TEMP = CONSTANT(EXTRACT(!ISERROR(W1:W3), W1:W3))</code>
Erstellt drei neue Spalten TEMP, WX und WY. Die drei Spalten enthalten die gültigen Zeilen (d. h. Zeilen, die keine ???-Zellen enthalten) von Spalte W1 bis W3. Die Einfügung der Makrofunktion EXTRACT innerhalb von CONSTANT verhindert, dass die Spalten TEMP, WX und WY neu berechnet werden, wenn sich die Spalten W1 bis W3 ändern, und beugt so der rechenintensiven Ausführung der Makrofunktion EXTRACT vor.

Zugehörige Funktionen

Funktion	Beschreibung
BUFFER	Kopiert den Eingabebereich und aktualisiert die Werte dynamisch.

COS

Syntax

`COS(Daten [, Einheit])`

Parameter

Daten

Die numerischen Werte, deren Kosinus berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Einheit

Dieses optionale Schlüsselwort legt fest, ob die Eingabewerte und Ergebnisse als Grad oder als Radiant interpretiert werden. Wählen Sie eine der folgenden Angaben aus:

RADIAN - Führt die Berechnungen in Radiant aus (Standard)

DEGREE - Führt die Berechnungen in Grad aus

Wird dieser Parameter nicht angegeben, ist der Standard Radiant. (Die Umrechnung von Radiant in Grad erfolgt durch Division durch Pi und Multiplikation mit 180.)

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

COS berechnet den Kosinus der Werte im angegebenen Datenbereich. COS gibt für jede Eingabespalte eine neue Spalte zurück, die den Kosinus der Zahlen in der entsprechenden Eingabespalte enthält.

Beispiele

$TEMP = COS(PI)$ oder $TEMP = COS(PI, 0)$ oder $TEMP = COS(PI, RADIAN)$ Gibt eine neue Spalte TEMP zurück, die den Einzelwert -1 enthält.
$TEMP = COS(90, 1)$ oder $TEMP = COS(90, DEGREE)$ Gibt eine neue Spalte TEMP zurück, die den Einzelwert null enthält.
$TEMP = COS(W1)$ oder $TEMP = COS(W1, 0)$ oder $TEMP = COS(W1, RADIAN)$ Erstellt eine neue Spalte TEMP, in der jeder Wert den Kosinus (in Radiant) des Inhalts von Spalte W1 darstellt.
$TEMP = COS(W1:W3, 1)$ Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der Kosinus des Inhalts von Spalte W1, die Werte von Spalte WX sind der Kosinus des Inhalts von Spalte W2 und die Werte von Spalte WY sind der Kosinus des Inhalts von Spalte W3. Alle Werte sind in Grad angegeben.
$TEMP = COS(W1[10:20])$ Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils den Kosinus der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
$TEMP = COS(W1[1:5]:W2)$ Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der Kosinus der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind der Kosinus der entsprechenden Zeilen von Spalte W2. Alle Werte sind in Radiant angegeben.

Zugehörige Funktionen

Funktion	Beschreibung
ACOS	Berechnet den Arkuskosinus des Inhalts des angegebenen Datenbereichs.
COSH	Berechnet den Hyperbelkosinus des Inhalts des angegebenen Datenbereichs.
SIN	Berechnet den Sinus des Inhalts des angegebenen Datenbereichs.
TAN	Berechnet den Tangens des Inhalts des angegebenen Datenbereichs.

COSH

Syntax

COSH(Daten [, Einheit])

Parameter

Daten

Die numerischen Werte, deren Hyperbelkosinus berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Einheit

Dieses optionale Schlüsselwort legt fest, ob die Eingabewerte und Ergebnisse als Grad oder als Radiant interpretiert werden. Wählen Sie eine der folgenden Angaben aus:

RADIAN - Führt die Berechnungen in Radiant aus (Standard)

DEGREE - Führt die Berechnungen in Grad aus

Wird dieser Parameter nicht angegeben, ist der Standard Radiant. (Die Umrechnung von Radiant in Grad erfolgt durch Division durch Pi und Multiplikation mit 180.)

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

COSH berechnet den Hyperbelkosinus der Werte im angegebenen Datenbereich. Bei x in Radiant ist der Hyperbelkosinus einer Zahl

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

Dabei ist e die natürliche Zahl, 2,7182818. COSH gibt für jede Eingabespalte eine neue Spalte zurück, die den Hyperbelkosinus der Zahlen in der entsprechenden Eingabespalte enthält.

Anmerkung: Wenn der Wert x zu groß ist, wird ein Überlauferfehler zurückgegeben. Dies geschieht, wenn $\cosh(x)$ den maximalen 32-Bit-Gleitkommawert überschreitet.

Beispiele

TEMP = COSH(0) oder TEMP = COSH(0, 0) oder TEMP = COSH(0, RADIAN)
Gibt eine neue Spalte TEMP zurück, die den Wert eins enthält.
TEMP = COSH(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert den Hyperbelkosinus (in Radiant) des Inhalts von Spalte W1 darstellt.
TEMP = COSH(W1:W3, 1) oder TEMP = COSH(W1:W3, DEGREE)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der Hyperbelkosinus des Inhalts von Spalte W1, die Werte von Spalte WX sind der Hyperbelkosinus des Inhalts von Spalte W2 und die Werte von Spalte WY sind der Hyperbelkosinus des Inhalts von Spalte W3. Alle Werte sind in Grad angegeben.
TEMP = COSH(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils den Hyperbelkosinus der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = COSH(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der Hyperbelkosinus der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind der Hyperbelkosinus der entsprechenden Zeilen von Spalte W2. Alle Werte sind in Radiant angegeben.

Zugehörige Funktionen

Funktion	Beschreibung
ACOS	Berechnet den Arkuskosinus des Inhalts des angegebenen Datenbereichs.
COS	Berechnet den Kosinus des Inhalts des angegebenen Datenbereichs.
SINH	Berechnet den Hyperbelsinus des Inhalts des angegebenen Datenbereichs.
TANH	Berechnet den Hyperbeltangens des Inhalts des angegebenen Datenbereichs.

COT

Syntax

COT(Daten [, Einheit])

Parameter

Daten

Die numerischen Werte, deren Kotangens berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Einheit

Dieses optionale Schlüsselwort legt fest, ob die Eingabewerte und Ergebnisse als Grad oder als Radiant interpretiert werden. Wählen Sie eine der folgenden Angaben aus:

RADIAN - Führt die Berechnungen in Radiant aus (Standard)

DEGREE - Führt die Berechnungen in Grad aus

Wird dieser Parameter nicht angegeben, ist der Standard Radiant. (Die Umrechnung von Radiant in Grad erfolgt durch Division durch Pi und Multiplikation mit 180.)

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

COT berechnet den Kotangens der Werte im angegebenen Datenbereich. Der Kotangens ist der Kehrwert des Tangens. COT gibt für jede Eingabespalte eine neue Spalte zurück, die den Kotangens der Zahlen in der entsprechenden Eingabespalte enthält.

Anmerkung: Wenn eine Zelle einen Wert enthält, dessen Tangens null ist, ist der Kotangens unendlich. In diesem Fall gibt COT die größte 32-Bit-Gleitkommazahl zurück.

Beispiele

TEMP = COT(90) oder TEMP = COT(90, 0) oder TEMP = COT(90, RADIAN)
Gibt eine neue Spalte TEMP zurück, die den Wert -0,5 enthält.
TEMP = COT(0)
Gibt eine neue Spalte TEMP zurück, die den Wert MAX_FLOAT_32 enthält.

TEMP = COT(W1, 1) oder TEMP = COT(W1, DEGREE)
Erstellt eine neue Spalte TEMP, in der jeder Wert den Kotangens des Inhalts (in Grad) von Spalte W1 darstellt.
TEMP = COT(W1:W3, 1)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der Kotangens des Inhalts von Spalte W1, die Werte von Spalte WX sind der Kotangens des Inhalts von Spalte W2 und die Werte von Spalte WY sind der Kotangens des Inhalts von Spalte W3. Alle Werte sind in Grad angegeben.
TEMP = COT(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils den Kotangens der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = COT(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der Kotangens der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind der Kotangens der entsprechenden Zeilen von Spalte W2. Alle Werte sind in Radiant angegeben.

Zugehörige Funktionen

Funktion	Beschreibung
ACOT	Berechnet den Arkuskotangens des Inhalts des angegebenen Datenbereichs.
COS	Berechnet den Kosinus des Inhalts des angegebenen Datenbereichs.
SIN	Berechnet den Sinus des Inhalts des angegebenen Datenbereichs.
TAN	Berechnet den Tangens des Inhalts des angegebenen Datenbereichs.

COUNT

Syntax

COUNT(Daten)

Parameter

Daten

Der Zellenbereich, in dem die Anzahl der Zellen gezählt werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

COUNT berechnet die Anzahl der Werte im angegebenen Datenbereich. COUNT gibt eine neue Spalte mit einem Einzelwert zurück, der die Anzahl der Zellen, die Werte enthalten, innerhalb des angegebenen Datenbereichs darstellt.

Anmerkung: Bei einer leeren Spalte wird null zurückgegeben.

Beispiele

TEMP = COUNT(AVG(W1:W5))
Erstellt eine neue Spalte TEMP, die den Einzelwert eins enthält (die Funktion AVG gibt im Standardmodus eine einzelne Zelle zurück).
TEMP = COUNT(W1)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Anzahl der Zellen mit Werten in Spalte W1 angibt.
TEMP = COUNT(W1:W3)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Anzahl der Zellen mit Werten in den Spalten W1, W2 und W3 angibt.
TEMP = COUNT(W1[10:20])
Erstellt eine neue Spalte TEMP, die den Wert 11 enthält (Bereiche schließen die Grenzen ein), sofern sämtliche Zellen Werte enthalten.
TEMP = COUNT(W1[1:5]:W4)
Erstellt eine neue Spalte TEMP, die den Wert 20 enthält (5 Zellen in jeder Spalte bei 4 Spalten ergeben 20 Zellen), sofern sämtliche Zellen Werte enthalten.
TEMP = COUNT(W1[1:10])
Erstellt eine neue Spalte TEMP, die den Wert 3 enthält, sofern die Zeilen 1-3 von Spalte W1 Werte enthalten und die Zeilen 4-10 leer sind.

Zugehörige Funktionen

Funktion	Beschreibung
COUNT_DIFF	Gibt jeden eindeutigen Wert in der Eingabe mit einer Zahl zurück, die die Häufigkeit des betreffenden Werts angibt.
OFFSET	Gibt den Offset jedes Werts in der Eingabespalte vom Anfang des Datenstroms an.
SUM oder TOTAL	Berechnet die Summe eines Zellenbereichs.

COUNT_DIFF



Syntax

COUNT_DIFF(Daten)

Parameter

Daten

Der Zellenbereich, für den die eindeutigen Werte und die Häufigkeit ihres Vorkommens gezählt werden sollen. Dabei kann es sich um einen konstanten Wert, eine

Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

COUNT_DIFF ermittelt die Anzahl verschiedener Werte im angegebenen Datenbereich und zählt, wie häufig jeder Wert vorkommt. COUNT_DIFF gibt zwei neue Spalten zurück. Die erste Spalte enthält jeden eindeutigen Wert. Die zweite Spalte gibt an, wie häufig der entsprechende Wert im Datenbereich vorkommt. COUNT_DIFF ignoriert beim Vergleich von Zeichenfolgen die Groß-/Kleinschreibung. So werden "Aaa", "aAa" und "AAA" alle derselben Klasse zugerechnet.

Anmerkung: Die Makrofunktion COUNT_DIFF benötigt unter Umständen lange Zeit für die Berechnung, wenn die Daten umfangreich sind. Es wird ein Fortschrittsleiste angezeigt, bis die Berechnung abgeschlossen ist. Wenn Sie die Berechnung abbrechen möchten, klicken Sie auf das "X" des Fortschrittsleistes und löschen Sie die Funktionsdefinition, die die Makrofunktion COUNT_DIFF enthält.

Beispiele

TEMP = COUNT_DIFF(COLUMN(1,2,3,1))	Erstellt zwei neue Spalten TEMP und WX. TEMP enthält die Werte 1, 2 und 3. WX enthält die Anzahlen 2, 1 und 1.
TEMP = COUNT_DIFF(COLUMN("x", "a", "a", "b"))	Erstellt zwei neue Spalten TEMP und WX. TEMP enthält die Werte x, a und b. WX enthält die Anzahlen 1, 2 und 1.
TEMP = COUNT_DIFF(W1)	Erstellt zwei neue Spalten TEMP und WX. TEMP enthält alle eindeutigen Werte in Spalte W1, WX enthält einen Zähler für jede entsprechende Zeile von TEMP.
TEMP = COUNT_DIFF(W1:W3)	Erstellt zwei neue Spalten TEMP und WX. TEMP enthält alle eindeutigen Werte in Spalte W1 bis W3, WX enthält einen Zähler für jede entsprechende Zeile von TEMP.
TEMP = COUNT_DIFF(W1[10:20])	Erstellt zwei neue Spalten TEMP und WX. TEMP enthält alle eindeutigen Werte in Zeile 10-20 von Spalte W1, WX enthält einen Zähler für jede entsprechende Zeile von TEMP.
TEMP = COUNT_DIFF(W1[1:5]:W4)	Erstellt zwei neue Spalten TEMP und WX. TEMP enthält alle eindeutigen Werte in Zeile 1-5 von Spalte W1 bis W4, WX enthält einen Zähler für jede entsprechende Zeile von TEMP.

Zugehörige Funktionen

Funktion	Beschreibung
COUNT	Zählt die Anzahl der Zellen, die Werte enthalten, im angegebenen Datenbereich.
HISTOGRAM	Berechnet das Histogramm eines angegebenen Datenbereichs mithilfe angegebener Bin-Grenzen.

COV



Syntax

COV(Daten1, Daten2)

Parameter

Daten1

Das erste Dataset. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Das zweite Dataset. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

COV zählt die Kovarianz der beiden angegebenen Datenbereiche.¹ Dabei werden so viele Spalten wie die minimale Dimension der Eingaben (Breite und Höhe) ausgegeben. COV wird wie folgt berechnet:

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

Dabei sind x und y Spalten, die dieselbe Anzahl von Werten N enthalten, μ_x ist der Durchschnitt von x und μ_y ist der Durchschnitt von y .

Beispiele

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

Erstellt eine neue Spalte TEMP, die den Wert 0,5 enthält.

```
TEMP = COV(W1, W2)
```

Erstellt eine neue Spalte TEMP, die die Kovarianz zwischen den Daten in Spalte W1 und den Daten in Spalte W2 enthält.


```
TEMP = COV(W1:W2, W3:W4)
```

Erstellt zwei neue Spalten TEMP und WX. Die Spalte TEMP enthält die Kovarianz zwischen den Spalten W1 und W3. Die Spalte WX enthält die Kovarianz zwischen den Spalten W2 und W4.

```
TEMP = COV(W1[1:50]:W2, W3[100:200]:W4)
```

Erstellt zwei neue Spalten TEMP und WX. Die Spalte TEMP enthält die Kovarianz zwischen Zeile 1-50 von Spalte W1 und Zeile 100-200 von Spalte W3. Die Spalte WX enthält die Kovarianz zwischen Zeile 1-50 von Spalte W2 und Zeile 100-200 von Spalte W4.

CURRENT_DATE

Syntax

```
CURRENT_DATE([Format])
```

Parameter

Format

Eines der Schlüsselwörter in der folgenden Tabelle, das das Datumsformat von Datumszeichenfolge angibt.

Beschreibung

CURRENT_DATE gibt das aktuelle Datum in Format zurück. Das Datum wird durch die Systemzeit auf dem IBM -Server festgelegt. Wenn kein Schlüsselwort Format angegeben wird, wird standardmäßig das Format BEGRENZ_M_T_J verwendet.

Bei allen empfohlenen Datenbanken versucht IBM Kampagne, das Makro CURRENT_DATE mithilfe eines von der Datenbank unterstützten SQL-Aufrufs für die aktuelle Uhrzeit (z. B. SYSDATE, GETDATE, DATE oder TODAY) in der Datenbank auszuführen. In diesen Fällen werden von dieser Makrofunktion alle Parameter (einschließlich des Datumsformats) ignoriert und die Ausgabe schließt alle Angaben ein, die von der Datenbank zurückgegeben werden (unter Umständen kann die Ausgabe also eine Zeitkomponente enthalten). Falls dies der Fall ist und nur das Datum oder das Datum in einem anderen Format zurückgegeben werden soll, können Sie mit Raw-SQL ein benutzerdefiniertes Makro schreiben oder IBM -Makros verwenden. Beispiel:

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

In einigen Fällen wird auf dem IBM Kampagne-Server das Makro CURRENT_DATE() ausgeführt (nämlich wenn eine unstrukturierte Datei oder eine nicht empfohlene Datenbank ohne entsprechende SQL-Unterstützung vorliegt oder wenn der Campaign-Makroausdruck in der Datenbank nicht aufgelöst werden kann). In diesen Fällen werden alle Parameter erkannt und die Ausgabe wird im ausgewählten Format zurückgegeben.

Anmerkung: Weitere Informationen zu den verfügbaren Datumsformaten finden Sie unter "Gültige Formatschlüsselwörter".

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

Wenn das aktuelle Datum der 7. September 2000 ist, gibt CURRENT_DATE() "09/07/00" zurück.

Zugehörige Funktionen

Funktion	Beschreibung
DATE_FORMAT	Konvertiert Datumsangaben von einem Format in ein anderes.
DATE_JULIAN	Gibt das julianische Datum der Eingabe zurück.
DATE_STRING	Gibt die Datumszeichenfolge des julianischen Datums zurück.
DATE	Konvertiert eine Datumszeichenfolge in ein julianisches Datum.

CURRENT_DAY

Syntax

CURRENT_DAY()

Beschreibung

CURRENT_DAY gibt den aktuellen Tag des Monats als Zahl zwischen 1 und 31 zurück. Das Datum wird durch die Systemzeit auf dem IBM -Server festgelegt.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

Wenn das aktuelle Datum der 19. Juni ist, gibt CURRENT_DAY() die Zahl 19 zurück.

Zugehörige Funktionen

Funktion	Beschreibung
CURRENT_JULIAN	Gibt die julianische Zahl für das aktuelle Datum zurück.
CURRENT_MONTH	Gibt den aktuellen Monat als Zahl zurück.
CURRENT_TIME	Gibt die aktuelle Uhrzeit als Zeichenfolge zurück.
CURRENT_WEEKDAY	Gibt den aktuellen Wochentag als Zahl zurück.
CURRENT_YEAR	Gibt das aktuelle Jahr als Zahl zurück.

CURRENT_JULIAN

Syntax

CURRENT_JULIAN()

Beschreibung

CURRENT_JULIAN() gibt die julianische Zahl für das aktuelle Datum (die Anzahl der seit dem 1. Januar 0000 vergangenen Tage) zurück. CURRENT_JULIAN() ist äquivalent zu dem Makro DATE(CURRENT_DATE()).

Beispiele

Wenn das aktuelle Datum der 31. August 2000 ist, gibt CURRENT_JULIAN() die Zahl 730729 zurück.

Zugehörige Funktionen

Funktion	Beschreibung
CURRENT_DAY	Gibt den aktuellen Tag als Zahl zurück.
CURRENT_MONTH	Gibt den aktuellen Monat als Zahl zurück.
CURRENT_TIME	Gibt die aktuelle Uhrzeit als Zeichenfolge zurück.
CURRENT_WEEKDAY	Gibt den aktuellen Wochentag als Zahl zurück.
CURRENT_YEAR	Gibt das aktuelle Jahr als Zahl zurück.

CURRENT_MONTH

Syntax

CURRENT_MONTH()

Beschreibung

CURRENT_MONTH gibt den aktuellen Monat des Jahres als Zahl zwischen 1 und 12 zurück.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

Wenn das aktuelle Datum der 19. Juni ist, gibt CURRENT_MONTH() die Zahl 6 zurück.

Zugehörige Funktionen

Funktion	Beschreibung
CURRENT_DAY	Gibt den aktuellen Tag als Zahl zurück.
CURRENT_JULIAN	Gibt die aktuelle julianische Zahl zurück.
CURRENT_TIME	Gibt die aktuelle Uhrzeit als Zeichenfolge zurück.
CURRENT_WEEKDAY	Gibt den aktuellen Wochentag als Zahl zurück.
CURRENT_YEAR	Gibt das aktuelle Jahr als Zahl zurück.

CURRENT_TIME

Syntax

CURRENT_TIME()

Beschreibung

CURRENT_TIME gibt die aktuelle Uhrzeit als Zeichenfolge zurück. Die Uhrzeit wird durch die Systemzeit auf dem IBM -Server festgelegt.

Datumseinstellung in Ihrer Webanwendung

Damit Datumsangaben in Ihrer Webanwendung bei aktuellen Versionen von IBM Kampagne korrekt angezeigt werden, muss die Konfigurationsdatei Ihres Back-End-Servers ordnungsgemäß konfiguriert werden. Dies ist besonders wichtig bei den Parametern `dDateFormat` und `DateOutputFormatString` für die Datenbank, die die Systemtabellen enthält. Wenn diese nicht ordnungsgemäß konfiguriert sind, werden Datumsangaben auch in Campaign falsch angezeigt. Diese Eigenschaften können mithilfe von IBM Marketing Platform konfiguriert werden.

Sprachspezifische Datumsangaben in Ihrer Webanwendung

Anmerkung: Alle referenzierten Dateien werden vom Installationsprogramm der Webanwendung installiert, sofern nicht anders angegeben.

Wichtig: `webapphome` steht für das Verzeichnis, in dem die Campaign-Webanwendung installiert wurde. `Sprachcode` steht für die Spracheinstellung(en), die Sie für Ihr System festlegen.

1. Bearbeiten Sie die Datei `webapphome/conf/campaign_config.xml`, um sicherzustellen, dass `Sprachcode` in der durch Kommas getrennten Liste im Tag `<supportedLocales>` enthalten ist, wie im Folgenden gezeigt:
`<supportedLocales>en_US, Sprachcode</supportedLocales>`
2. Kopieren Sie im Verzeichnis `webapphome/webapp` die gesamte Verzeichnisstruktur von `en_US` nach `Sprachcode` (Groß-/Kleinschreibung muss beachtet werden).
3. Kopieren Sie in `webapphome/webapp/WEB-INF/classes/resources` `StaticMessages_en_US.properties` nach `StaticMessages_Sprachcode.properties`. Kopieren Sie außerdem `ErrorMessages_en_US.properties` nach `ErrorMessages_Sprachcode.properties`.
4. Bearbeiten Sie `StaticMessages_Sprachcode.properties`: Suchen Sie nach `DatePattern` und ändern Sie den Eintrag in `DatePattern=dd/MM/yyyy` (Groß-/Kleinschreibung muss beachtet werden).

Anmerkung: Dieses Format wird von Java™ definiert. Vollständige Informationen zu dem Format finden Sie in der Java-Dokumentation für `java.text.SimpleDateFormat` unter <http://java.sun.com>. Die Datei `StaticMessages.properties` braucht nicht geändert zu werden.

5. Bei WebSphere: Erstellen Sie eine neue JAR-Datei der Webanwendung.
6. Bei WebLogic: Entfernen Sie das aktuelle Modul der Webanwendung.
 - a. Fügen Sie das neue Modul hinzu.
 - b. Führen Sie die Bereitstellung der Webanwendung erneut aus.
 - c. Ein Neustart des Kampagne-Listeners ist nicht erforderlich.
7. Stellen Sie sicher, dass `Sprachcode` in der Spracheinstellung des Web-Browsers die höchste Priorität hat. Weitere Informationen finden Sie in den folgenden

Abschnitten, die die korrekte Spracheinstellung für Ihren Web-Browser und die Anzeige einer bestimmten Sprache auf Ihrem Computer behandeln.

Anmerkung: Achten Sie darauf, in Sprachcode einen Bindestrich und keinen Unterstrich zu verwenden. Die Konfiguration der Webanwendung ist der einzige Ort, an dem anstelle eines Unterstrichs ein Bindestrich verwendet wird.

8. Melden Sie sich bei Campaign an. Datumsangaben sollten in Campaign in dem Format angezeigt werden, das Sie in `StaticMessages_Sprachcode.properties` angegeben haben.

Informationen zur Konfiguration der Uhrzeit für IBM Kampagne finden Sie in der Dokumentation zu *IBM Kampagne*.

Beispiele

Wenn die Uhrzeit 10:54 ist, gibt `CURRENT_TIME()` die Zeichenfolge "10:54:00" zurück.

Zugehörige Funktionen

Funktion	Beschreibung
<code>CURRENT_DAY</code>	Gibt den aktuellen Tag als Zahl zurück.
<code>CURRENT_JULIAN</code>	Gibt die aktuelle julianische Zahl zurück.
<code>CURRENT_WEEKDAY</code>	Gibt den aktuellen Wochentag als Zahl zurück.
<code>CURRENT_YEAR</code>	Gibt das aktuelle Jahr als Zahl zurück.

CURRENT_WEEKDAY

Syntax

`CURRENT_WEEKDAY()`

Beschreibung

`CURRENT_WEEKDAY` gibt den aktuellen Wochentag als Zahl zwischen 0 und 6 zurück. Der Sonntag wird als 0 dargestellt, der Montag als 1 usw.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

Wenn der aktuelle Wochentag ein Freitag ist, gibt `CURRENT_WEEKDAY()` die Zahl 5 zurück.

Zugehörige Funktionen

Funktion	Beschreibung
<code>CURRENT_DAY</code>	Gibt den aktuellen Tag als Zahl zurück.
<code>CURRENT_JULIAN</code>	Gibt die aktuelle julianische Zahl zurück.
<code>CURRENT_MONTH</code>	Gibt den aktuellen Monat als Zahl zurück.
<code>CURRENT_TIME</code>	Gibt die aktuelle Uhrzeit als Zeichenfolge zurück.

Funktion	Beschreibung
CURRENT_YEAR	Gibt das aktuelle Jahr als Zahl zurück.

CURRENT_YEAR

Syntax

CURRENT_YEAR()

Beschreibung

CURRENT_YEAR gibt das aktuelle Jahr als Zahl zurück.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

Wenn das aktuelle Jahr 2000 ist, gibt CURRENT_YEAR() die Zahl 2000 zurück.

Zugehörige Funktionen

Funktion	Beschreibung
CURRENT_DAY	Gibt den aktuellen Tag als Zahl zurück.
CURRENT_JULIAN	Gibt die aktuelle julianische Zahl zurück.
CURRENT_MONTH	Gibt den aktuellen Monat als Zahl zurück.
CURRENT_TIME	Gibt die aktuelle Uhrzeit als Zeichenfolge zurück.
CURRENT_WEEKDAY	Gibt den aktuellen Wochentag als Zahl zurück.
MONTHOF	Gibt den Monat des Jahres als Zahl zurück.
WEEKDAYOF	Gibt den Wochentag der Woche als Zahl zurück.
YEAROF	Gibt das Jahr als Zahl zurück.

CV_FOLDS



Syntax

CV_FOLDS(Anzahl_Teilmengen, Daten [, Klassendaten] [Seed])

Parameter

Anzahl_Teilmengen

Die Anzahl der Teilmengen, die für die Kreuzvalidierung erstellt werden sollen. Dieser Wert muss eine positive Ganzzahl größer als 1 sein. Der Wert muss kleiner als 65.536 oder die Anzahl der Zeilen in Daten sein, je nachdem, welche Zahl kleiner ist.

Daten

Die Eingabevariablen. Dabei kann es sich um eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Klassendaten

Wenn dieser optionale Datenbereich angegeben wird, erstellt die Makrofunktion CV_FOLDS Teilmengen und sorgt dabei für gleichmäßige Klassenwahrscheinlichkeiten. Der Inhalt von Klassendaten dient als Ausgabe für das jeweilige Eingabemuster.

Wenn es sich bei Klassendaten um eine einzelne Spalte handelt, setzt CV_FOLDS voraus, dass die angegebene Spalte Werte für mehrere Ausgabeklassen enthält (d. h., jeder unterschiedliche Wert wird als separate Klasse eingestuft). Wenn es sich bei Klassendaten um einen Datenbereich handelt, wird jede Ausgabespalte als unterschiedliche Klasse eingestuft. (Bei einem Datenbereich sind die Werte in jeder Spalte eins, wenn ein Muster der betreffenden Klasse angehört, oder null, wenn das Muster der betreffenden Klasse nicht angehört.)

Informationen zur Formatdefinition von Klassendaten (identisch mit Daten) finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Seed

Ein Basiswert für den Zufallsgenerator. Der Wert muss eine ganze Zahl sein.

Beschreibung

CV_FOLDS unterteilt die Eingabedaten gleichmäßig in die angegebene Anzahl von Teilmengen. Jede Teilmenge enthält dieselbe Anzahl von Eingabemustern.² Jede Zeile des Eingabedatenbereichs wird in einer Teilmenge platziert, indem eine neue Spalte zurückgegeben wird, die Teilmengetnummern zwischen eins und Anzahl_Teilmengen enthält.

Wenn der optionale Parameter Klassendaten angegeben wird, werden aufgrund der Angaben zur Ausgabeklasse Teilmengen für die Kreuzvalidierung erstellt, wobei die Ausgabeklassenwahrscheinlichkeiten berücksichtigt werden. Innerhalb jeder Teilmenge ist die Wahrscheinlichkeit der jeweiligen Ausgabeklasse also identisch.³

Beispiele

```
TEMP = CV_FOLDS(3, W1, 0)
```

Erstellt eine neue Spalte TEMP, die für jede Zeile von Spalte W1 einen Wert enthält. Die Spalte TEMP enthält die Werte 1, 2 und 3 für die drei verschiedenen Teilmengen. Es werden keine Klassenwahrscheinlichkeiten berücksichtigt. Der Wert null wird als Generierungswert für den Zufallsgenerator verwendet.

```
TEMP = CV_FOLDS(100, W1:W15)
```

Erstellt eine neue Spalte TEMP, die für jede Zeile der kürzesten Spalte in W1-W15 einen Wert enthält. Die Spalte TEMP enthält die Werte 1-100 für die 100 verschiedenen Teilmengen. Es werden keine Klassenwahrscheinlichkeiten berücksichtigt. Es wird ein zufälliger Anfangswert ausgewählt.

```
TEMP = CV_FOLDS(50, W1:W10, W11)
```

Erstellt eine neue Spalte TEMP, die für jede Zeile der kürzesten Spalte in W1-W10 einen Wert enthält. Die Spalte TEMP enthält die Werte 1-50 für die 50 verschiedenen Teilmengen. Die Spalte W11 enthält die Ausgabeklassen. Für jede Teilmenge gelten dieselben Ausgabeklassenwahrscheinlichkeiten. Es wird ein wahlfreier Generierungswert ausgewählt.

```
TEMP = CV_FOLDS(10, W1:W10, W11:W15, 96)
```

Erstellt eine neue Spalte TEMP, die für jede Zeile der kürzesten Spalte in W1-W10 einen Wert enthält. Die Spalte TEMP enthält die Werte 1-10 für die 10 verschiedenen Teilmengen. Jede der Ausgabespalten W11-W15 stellt eine Ausgabeklasse dar. Für jede Teilmenge gelten dieselben Ausgabeklassenwahrscheinlichkeiten. Der Wert 96 wird als Generierungswert für den Zufallsgenerator verwendet.

DATALINK



Syntax

```
DATALINK([Arbeitsblatt,] Zellen)
```

Parameter

Arbeitsblatt

Der Name des Arbeitsblattes in der aktuellen IBM PredictiveInsight-Arbeitssitzung, zu dem eine Verknüpfung hergestellt werden soll (z. B. Blatt1). Wenn kein Wert angegeben ist, wird das aktuelle Arbeitsblatt verwendet.

Zellen

Die konkreten Zellen innerhalb von Arbeitsblatt, zu denen eine Verknüpfung hergestellt werden soll. Der Parameter Zellen gibt die Spalte(n) und Zeile(n) an, die für die Verknüpfung vorgesehen sind, und kann auf eine der folgenden Arten angegeben werden:

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

Die Zeichen C und R geben die Spalte (column) bzw. die Zeile (row) an. Die Variablen n, m, N und M sind die Zeilen- und Spaltennummern.

Beschreibung

DATALINK erstellt einen internen Link zu Daten in der aktuellen IBM PredictiveInsight-Arbeitssitzung. DATALINK gibt die Anzahl der Spalten zurück, die im Parameter Zellen angegeben ist. Änderungen in den Quelldatenspalten werden automatisch übernommen.

Beispiele

```
TEMP = DATALINK(S1)
```

Erstellt eine neue Spalte TEMP, die die Werte in Spalte 1 des aktuellen Arbeitsblattes enthält.

TEMP = DATALINK(Blatt2, C1:C3) Erstellt drei neue Spalten TEMP, WX und WY, die die Werte in Spalte 1-3 des Arbeitsblattes Blatt2 enthalten.
TEMP = DATALINK(Blatt4, C5R10) oder TEMP = DATALINK(Blatt4, R10C5) Erstellt eine neue Spalte TEMP, die den Zellenwert in Spalte 5, Zeile 10 des Arbeitsblattes Blatt4 enthält.
TEMP = DATALINK(Blatt1, C1R1:C3R500) oder TEMP = DATALINK(Blatt1, R1C1:R500C3) Erstellt drei neue Spalten TEMP, WX und WY, die die Werte in Zeile 1-500 von Spalte 1-3 des Arbeitsblattes Blatt1 enthalten.

Zugehörige Funktionen

Funktion	Beschreibung
DDELINK	Erstellt einen externen Link zu Daten aus einer anderen Windows-Anwendung.

DATE

Syntax

DATE(Datumszeichenfolge [, Format])

Parameter

Datumszeichenfolge

Ein Text, der ein gültiges Datum darstellt.

Format

Eines der Schlüsselwörter in der Tabelle unter "Gültige Formatschlüsselwörter", das das Datumsformat von Datumszeichenfolge angibt.

Beschreibung

DATE konvertiert eine Datumszeichenfolge in ein julianisches Datum (die Anzahl der seit dem 1. Januar 0000 vergangenen Tage). Über das optionale Schlüsselwort Format, das die Darstellung des Datums festlegt, wird nahezu jedes beliebige Datumsformat unterstützt. Wenn kein Schlüsselwort Format angegeben wird, wird standardmäßig das Format BEGRENZ_M_T_J verwendet.

Datumsformate weisen entweder eine feste Breite (dann wird der 28. Februar 1970 im Format MMTTJJJJ als 02281970 dargestellt) oder einen Begrenzer auf (dann kann der 28. Februar 1970 etwa als 2-28-1970 oder 02/28/1970 dargestellt werden). Alle vorherigen Beispiele sind Varianten des Formats BEGRENZ_M_T_JJ.

In begrenzten Formaten kann der Begrenzer ein Schrägstrich (/), ein Strich (-), ein Leerzeichen (), ein Komma (,) oder ein Doppelpunkt (:) sein; Jahre können 2- oder 4-stellig dargestellt werden; Monate können ausgeschrieben (z. B. Februar), abgekürzt (z. B. Feb) oder numerisch dargestellt werden (z. B. 2 oder 02).

Für alle zweistellig angegebenen Jahre gilt Folgendes:

- Zweistellige Jahresangaben, die kleiner als der Grenzwert für das Jahr 2000 sind (standardmäßig 20, kann jedoch vom Benutzer definiert werden), werden dem 21. Jahrhundert zugerechnet.
- Zweistellige Jahresangaben, die größer-gleich dem Grenzwert sind, werden dem 20. Jahrhundert zugerechnet.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

DATE("8/31/2000") gibt die Zahl 730729 zurück.

Gültige Formatschlüsselwörter

Schlüsselwort	Beschreibung	Beispiel(e)
MM	2-stelliger Monat	01, 02, 03, ..., 12
MMTT	2-stelliger Monat und 2-stelliger Tag	Der 31. März ist 0331
MMTTJJ	2-stelliger Monat, 2-stelliger Tag und 2-stelliges Jahr	Der 31. März 1970 ist 033170
MMTTJJJJ	2-stelliger Monat, 2-stelliger Tag und 4-stelliges Jahr	Der 31. März 1970 ist 03311970
BEGRENZ_M_T	Beliebiger Monat mit Begrenzer, gefolgt vom Tag	Der 31. März ist 3/31 oder 03-31
BEGRENZ_M_T_J	Belieb. Monat, Tag und Jahr, getrennt durch Begrenzer	März 31 1970 oder 3/31/70
BEGRENZ_J_M	Beliebiges Jahr mit Begrenzer, gefolgt vom Monat	70 März, 70-3 oder 1970/3
BEGRENZ_J_M_T	Belieb. Jahr, Monat und Tag, getrennt durch Begrenzer	1970 Mär 31 oder 70/3/31
JJMMM	2-stelliges Jahr und 3-stelliger Monat	70MÄR
JJMMTT	2-stelliges Jahr, 3-stelliger Monat und 2-stelliger Tag	70MÄR31
JJ	2-stelliges Jahr	70
JJMM	2-stelliges Jahr und 2-stelliger Monat	7003
JJMMTT	2-stelliges Jahr, 2-stelliger Monat und 2-stelliger Tag	700331
JJJJMMM	4-stelliges Jahr und 3-stelliger Monat	1970MÄR
JJJJMMTT	4-stelliges Jahr, 3-stelliger Monat und 2-stelliger Tag	1970MÄR31
JJJJ	4-stelliges Jahr	1970
JJJJMM	4-stelliges Jahr und 2-stelliger Monat	197003
JJJJMMTT	4-stelliges Jahr, 2-stelliger Monat und 2-stelliger Tag	19700331

BEGRENZ_M_J	Beliebiger Monat mit Begrenzer, gefolgt vom Jahr	3-70, 3/70, Mär 70, März 1970
BEGRENZ_T_M	Beliebiger Tag mit Begrenzer, gefolgt vom Monat	31-3, 31/3, 31 März
BEGRENZ_T_M_J	Belieb. Tag, Monat und Jahr, getrennt durch Begrenzer	31-MÄR-70, 31/3/1970, 31 03 70
TT	2-stelliger Tag	31
TTMMM	2-stelliger Tag und 3-stelliger Monat	31MÄR
TTMMMJJ	2-stelliger Tag, 3-stelliger Monat und 2-stelliges Jahr	31MÄR70
TTMMMJJJJ	2-stelliger Tag, 3-stelliger Monat und 4-stelliges Jahr	31MÄR1970
TTMM	2-stelliger Tag und 2-stelliger Monat	3103
TTMMJJ	2-stelliger Tag, 2-stelliger Monat und 2-stelliges Jahr	310370
TTMMJJJJ	2-stelliger Tag, 2-stelliger Monat und 4-stelliges Jahr	31031970
MMJJ	2-stelliger Monat und 2-stelliges Jahr	0370
MMJJJJ	2-stelliger Monat und 4-stelliges Jahr	031970
MMM	3-stelliger Monat	MÄR
MMMTT	3-stelliger Monat und 2-stelliger Tag	MÄR31
MMMTTJJ	3-stelliger Monat, 2-stelliger Tag und 2-stelliges Jahr	MÄR3170
MMMTTJJJJ	3-stelliger Monat, 2-stelliger Tag und 4-stelliges Jahr	MÄR311970
MMMJJ	3-stelliger Monat und 2-stelliges Jahr	MÄR70
MMMJJJJ	3-stelliger Monat und 4-stelliges Jahr	MÄR1970
MONTH	Monat des Jahres	Januar, Februar, März usw. oder Jan, Feb, Mär usw.
WEEKDAY	Wochentag	Sonntag, Montag, Dienstag usw. (Sonntag = 0)
WKD	Abgekürzter Wochentag	Son, Mon, Die usw. (Son = 0)

Zugehörige Funktionen

Funktion	Beschreibung
DATE_FORMAT	Konvertiert Datumsangaben von einem Format in ein anderes.
DATE_JULIAN	Gibt das julianische Datum der Eingabe zurück.

Funktion	Beschreibung
DATE_STRING	Gibt die Datumszeichenfolge des julianischen Datums zurück.
CURRENT_DATE	Gibt das aktuelle Datum in einem bestimmten Format zurück.

DATE_FORMAT

Syntax

DATE_FORMAT(Datumszeichenfolge, Eingabeformat, Ausgabeformat)

Parameter

Datumszeichenfolge

Ein Text, der ein gültiges Datum darstellt.

Eingabeformat

Eines der Schlüsselwörter in der folgenden Tabelle, das das Datumsformat von Datumszeichenfolge angibt.

Ausgabeformat

Eines der Schlüsselwörter in der folgenden Tabelle, das das vorgesehene Ausgabeformatsformat angibt.

Beschreibung

DATE_FORMAT() wandelt ein Datum von Eingabeformat in ein anderes Ausgabeformat um.

Wenn das Datum eine feste Breite aufweist, muss dem Format einer der folgenden Werte zugewiesen werden:

- TTMMJJ[JJ]
- TTMMMJJ[JJ]
- MMTTJJ[JJ]
- MMMTTJJ[JJ]
- JJ[JJ]MMTT
- JJ[JJ]MMMTT

MM ist eine aus 2 Ziffern, MMM eine aus 3 Buchstaben bestehende Abkürzung des Monats.

Wenn das Datum Begrenzer aufweist (als Begrenzer können Leerzeichen, Gedankenstriche und Schrägstriche verwendet werden), muss dem Format einer der folgenden Werte zugewiesen werden:

- BEGRENZ_T_M_J
- BEGRENZ_M_T_J
- BEGRENZ_J_M_T

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

DATE_FORMAT("012171", MMTJJ, MMTJJJJ) gibt die Zeichenfolge "01211971" zurück.

Anmerkung: Weitere Informationen zu gültigen Datumsformaten finden Sie unter „DATE“ auf Seite 77.

Zugehörige Funktionen

Funktion	Beschreibung
DATE	Konvertiert eine Datumszeichenfolge in ein julianisches Datum.
DATE_JULIAN	Gibt das julianische Datum der Eingabe zurück.
DATE_STRING	Gibt die Datumszeichenfolge des julianischen Datums zurück.

DATE_JULIAN

Syntax

DATE_JULIAN(Jahr, Monat, Tag)

Parameter

Jahr

Gültige 2-stellige oder 4-stellige Jahreszahl.

Monat

Gültige Monatszahl zwischen 1 und 12.

Tag

Gültige Tageszahl zwischen 1 und 31.

Beschreibung

DATE_JULIAN gibt das julianische Datum der angegebenen Eingabe zurück. Das julianische Datum ist die Anzahl der seit dem 1. Januar 0000 vergangenen Tage.

Beispiele

DATE_JULIAN (2000,08,31) gibt die Zahl 730729 zurück.

Zugehörige Funktionen

Funktion	Beschreibung
DATE	Konvertiert eine Datumszeichenfolge in ein julianisches Datum.

Funktion	Beschreibung
DATE_FORMAT	Konvertiert Datumsangaben von einem Format in ein anderes.
DATE_STRING	Gibt die Datumszeichenfolge des julianischen Datums zurück.

DATE_STRING

Syntax

```
DATE_STRING(julianisches_Datum [, 'Ausgabeformat'[, maximale_Länge]])
DATE_STRING(julianisches_Datum [, 'Formatierzeichenfolge' [,
maximale_Länge]])
```

Parameter

julianisches_Datum

Eine Zahl, die ein julianisches Datum darstellt, d. h. die Anzahl der seit dem 1. Januar 0000 vergangenen Tage.

Ausgabeformat

Zeichenfolge, gültiges Datumsformat.

maximale_Länge

Formatierzeichenfolge

Eine Formatierzeichenfolge, die optional eine beliebige Kombination der folgenden Formatcodes enthalten kann:

Code	Beschreibung
%a	Abgekürzter Wochentagsname
%A	Vollständiger Wochentagsname
%b	Abgekürzter Monatsname
%B	Vollständiger Monatsname
%c	Datums- und Uhrzeitdarstellung gemäß der Ländereinstellung
%d	Tag des Monats (01 - 31)
%H	Stunde im 24-Stunden-Format (00 - 23)
%I	Stunde im 12-Stunden-Format (01 - 12)
%j	Tag des Jahres (001 - 366)
%m	Monat (01 - 12)
%M	Minute (00 - 59)
%p	AM/PM-Anzeiger für 12-Stunden-Format gemäß der aktuellen Ländereinstellung
%S	Sekunde (00 - 59)
%U	Woche des Jahres, wobei Sonntag als erster Wochentag zählt (00 - 51)

Code	Beschreibung
%w	Wochentag (0 - 6; Sonntag ist 0)
%W	Woche des Jahres, wobei Montag als erster Wochentag zählt (00 - 51)
%x	Datumsdarstellung gemäß der aktuellen Ländereinstellung
%X	Uhrzeitdarstellung gemäß der aktuellen Ländereinstellung
%y	2-stelliges Jahr (00 - 99)
%Y	4-stelliges Jahr
%z, %Z	Name oder Abkürzung der Zeitzone; keine Ausgabe, falls Zeitzone unbekannt ist
%%	Prozentzeichen

Beschreibung

DATE_STRING gibt die Datumszeichenfolge des julianischen Datums zurück. Wenn Ausgabeformat nicht angegeben ist, wird das Standardschlüsselwort BEGRENZ_M_T_J verwendet.

Beispiele

DATE_STRING(730729) gibt die Zeichenfolge "08/31/00" zurück.

Anmerkung: Weitere Informationen zu gültigen Datumsformaten finden Sie unter „DATE“ auf Seite 77.

Zugehörige Funktionen

Funktion	Beschreibung
DATE	Konvertiert eine Datumszeichenfolge in ein julianisches Datum.
DATE_JULIAN	Gibt das julianische Datum der Eingabe zurück.
DATE_FORMAT	Konvertiert Datumsangaben von einem Format in ein anderes.

DAY_BETWEEN

Syntax

DAY_BETWEEN(Anfangsdatumszeichenfolge, Enddatumszeichenfolge [, Eingabeformat])

Parameter

Anfangsdatumszeichenfolge

Ein Text, der ein gültiges Datum darstellt, von dem aus die Anzahl der verstrichenen Tage berechnet werden soll.

Enddatumszeichenfolge

Ein Text, der ein gültiges Datum darstellt, bis zu dem die Anzahl der Tage berechnet werden soll. Dieses Datum muss dasselbe Format wie Anfangsdatumszeichenfolge aufweisen.

Eingabeformat

Eines der Schlüsselwörter in der folgenden Tabelle, das das Datumsformat von Anfangsdatumszeichenfolge und Enddatumszeichenfolge angibt.

Beschreibung

DAY_BETWEEN gibt die Anzahl der Tage zwischen Anfangsdatumszeichenfolge und Enddatumszeichenfolge zurück. Wenn Eingabeformat nicht angegeben ist, wird das Standardschlüsselwort BEGRENZ_M_T_J verwendet.

Beispiele

DAY_BETWEEN("08/25/00", "08/31/00") gibt die Zahl 6 zurück.

Anmerkung: Weitere Informationen zu gültigen Datumsformaten finden Sie unter „DATE“ auf Seite 77.

Zugehörige Funktionen

Funktion	Beschreibung
DAY_FROMNOW	Gibt die Anzahl der Tage vom aktuellen Datum bis zu einem angegebenen Datum zurück.
DAY_INTERVAL	Gibt die Anzahl der Tage zwischen zwei angegebenen Terminen zurück.

DAY_FROMNOW

Syntax

DAY_FROMNOW(Endjahr, Endmonat, Endtag)

Parameter

Endjahr

Gültige 2-stellige oder 4-stellige Jahreszahl.

Endmonat

Gültige Monatszahl zwischen 1 und 12.

Endtag

Gültige Tageszahl zwischen 1 und 31.

Beschreibung

DAY_FROMNOW gibt die Anzahl der Tage zwischen dem aktuellen Tag und dem durch Endjahr/Endmonat/Endtag angegebenen Datum zurück.

Anmerkung: Wenn das angegebene Datum in der Vergangenheit liegt, ist der Rückgabewert negativ.

Beispiele

Wenn das aktuelle Datum der 31. August 2000 ist, gibt DAY_FROMNOW(2000,12,31) die Zahl 122 zurück.

Zugehörige Funktionen

Funktion	Beschreibung
DAY_BETWEEN	Gibt die Anzahl der Tage zwischen zwei angegebenen Datumszeichenfolgen zurück.
DAY_INTERVAL	Gibt die Anzahl der Tage zwischen zwei angegebenen Terminen zurück.

DAY_INTERVAL

Syntax

DAY_INTERVAL(Anfangsjahr, Anfangsmonat, Anfangstag, Endjahr, Endmonat, Endtag)

Parameter

Anfangsjahr

Gültige 2-stellige oder 4-stellige Jahreszahl.

Anfangsmonat

Gültige Monatszahl zwischen 1 und 12.

Anfangstag

Gültige Tageszahl zwischen 1 und 31.

Endjahr

Gültige 2-stellige oder 4-stellige Jahreszahl.

Endmonat

Gültige Monatszahl zwischen 1 und 12.

Endtag

Gültige Tageszahl zwischen 1 und 31.

Beschreibung

DAY_INTERVAL gibt die Anzahl der Tage zwischen dem angegebenen Anfangsdatum (Anfangsjahr/Anfangsmonat/Anfangstag) und dem angegebenen Enddatum (Endjahr/Endmonat/Endtag) zurück.

Beispiele

DAY_INTERVAL(2000,8,31,2000,12,31) gibt die Zahl 122 zurück.

Zugehörige Funktionen

Funktion	Beschreibung
DAY_BETWEEN	Gibt die Anzahl der Tage zwischen zwei angegebenen Datumszeichenfolgen zurück.
DAY_FROMNOW	Gibt die Anzahl der Tage vom aktuellen Datum bis zu einem angegebenen Datum zurück.

DAYOF

Syntax

DAYOF(Datumszeichenfolge [, Eingabeformat])

Parameter

Datumszeichenfolge

Ein Text, der ein gültiges Datum darstellt.

Eingabeformat

Eines der Schlüsselwörter in der folgenden Tabelle, das das Datumsformat von Datumszeichenfolge angibt.

Beschreibung

DAYOF gibt für das durch Datumszeichenfolge angegebene Datum den Tag des Monats als Zahl zurück. Wenn Eingabeformat nicht angegeben ist, wird das Standard-schlüsselwort BEGRENZ_M_T_J verwendet.

Beispiele

DAYOF("08/31/00") gibt die Zahl 31 zurück.

Anmerkung: Weitere Informationen zu gültigen Datumsformaten finden Sie unter „DATE“ auf Seite 77.

DDELINK



Syntax

DDELINK(Service, Topic, Elemente)

Parameter

Service

Der Name des Service (z. B. excel), zu dem ein DDE-Link hergestellt werden soll.

Topic

Das Topic innerhalb von Service, zu dem der Link hergestellt werden soll. Bei den meisten Anwendungen ist das Topic ein Dateiname. Geben Sie den vollständigen Pfad und den Dateinamen des vorgesehenen Topics ein (z. B. c:\Bestand\Preise\05jan.xls).

Elemente

Die Elemente in Topic, zu denen der Link hergestellt werden soll. Die Syntax der Elemente hängt vom ausgewählten Service ab. Beispielsweise werden in Excel durch R1C1:R10C20 Zeile 1-10 und Spalte 1-20 ausgewählt.

Beschreibung

DDELINK erstellt einen DDE-Link (Dynamic Data Exchange) zu Daten in einer externen Windows-Anwendung. DDELINK gibt die Anzahl der Spalten zurück, die im Parameter Elemente angegeben ist. Änderungen in der Quellenanwendung werden im IBM PredictiveInsight-Arbeitsblatt automatisch übernommen.

Beispiele

TEMP = DDELINK(Excel, c:\excel\data.xls, C1:C2)
Erstellt zwei neue Spalten TEMP und WX, die die Werte in Spalte 1-2 des c:\excel\data.xls-Arbeitsblattes enthalten.
TEMP = DDELINK(Excel, c:\excel\data.xls, R1:R10)
Erstellt so viele neue Spalten, wie für die Aufnahme von Zeile 1-10 des c:\excel\data.xls-Arbeitsblattes erforderlich sind.
TEMP = DDELINK(Excel, c:\excel\data.xls, R1C1:R100C3)
Erstellt drei neue Spalten TEMP, WX und WY, die die Zeilen 1-100 des c:\excel\data.xls-Arbeitsblattes enthalten.
TEMP = DDELINK(123W, c:\lotsuite\sample.wk4, A:A1..A:C8)
Erstellt drei neue Spalten TEMP, WX und WY, die die Zeilen 1-8 von Spalte A-C des Arbeitsblattes A in der Lotus-Datei sample.wk4 enthalten.

Zugehörige Funktionen

Funktion	Beschreibung
DATALINK	Erstellt einen internen Link zu Daten in einem IBM PredictiveInsight-Arbeitsblatt.

DECIMATE



Syntax

DECIMATE(Spalte, Maximalwert)

Parameter

Spalte

Die Spalte der Werte, die verringert werden sollen. Alle Werte in dieser Spalte müssen positive Ganzzahlen kleiner als Maximalwert sein.

Maximalwert

Die Anzahl der Spalten, die zurückgegeben werden sollen. Es muss sich um eine positive Ganzzahl größer-gleich dem Maximalwert in Spalte handeln.

Beschreibung

DECIMATE konvertiert einen positiven Ganzzahlwert in ein binäres Muster, das Maximalwert Spalten lang ist. Wenn der Wert n ist, enthält die n -te Spalte eine Eins; alle anderen Spalten enthalten Nullen. Diese Makrofunktion gibt Maximalwert Spalten zurück.

Anmerkung: DECIMATE ist das Gegenteil der Makrofunktion MAXINDEX.

Beispiele

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

Erstellt drei neue Spalten TEMP, WX und WY mit einer Zeile für jede entsprechende Eingabezeile. Die erste Zeile enthält 1 0 0, die zweite enthält 0 1 0 und die dritte enthält 0 0 1.

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

Erstellt drei neue Spalten TEMP, WX und WY mit einer Zeile für jede entsprechende Eingabezeile. Die erste Zeile enthält 1 0 0, die zweite enthält 1 0 0 und die dritte enthält 0 1 0.

```
TEMP = DECIMATE(W1, 10)
```

Erstellt zehn neue Spalten mit einer Zeile für jede entsprechende Eingabezeile. Jede Zeile enthält in der Spalte, die den entsprechenden Eingabewert darstellt, eine einzelne Eins. Alle übrigen Spalten enthalten eine Null.

Zugehörige Funktionen

Funktion	Beschreibung
MAX	Berechnet das Maximum eines Zellenbereichs.
MAXINDEX	Gibt die Verzögerung der Eingabespaltenwerte um eine angegebene Anzahl von Zeitschritten an.

Funktion	Beschreibung
MIN	Berechnet das Minimum eines Zellenbereichs.

DELAY



Syntax

DELAY(Verzögerung, Daten)

Parameter

Verzögerung

Die Anzahl der für die Verzögerung vorgesehenen Zeitschritte. Dieser Wert muss eine positive Ganzzahl sein.

Daten

Die Werte, die verzögert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

DELAY gibt Werte im Eingabedatenbereich zurück, die um die angegebene Anzahl von Zeitschritten verzögert sind. Jede Eingabespalte wird als Datenserie in der Zeit betrachtet. DELAY gibt für jede Eingabespalte eine neue Spalte zurück. Jede neue Spalte enthält die zeitverzögerten Werte (die Anzahl der Zeitschritte wird durch Verzögerung angegeben) der Zahlen in der entsprechenden Eingabespalte.

Anmerkung: Die Makrofunktion DELAY gibt eine Spalte mit Werten zurück, sodass Zelle WY[x] = Daten[x + Verzögerung].

Anmerkung: Diese Funktion kann zum Erstellen von Mustern aus Zeitseriendaten eingesetzt werden. Um mehrere Verzögerungen zu erstellen, kann die Makrofunktion SLIDE_WINDOW verwendet werden.

Beispiele

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

Erstellt eine neue Spalte TEMP, die in Zelle 1-3 die Werte 2, 3 bzw. 4 enthält.

```
TEMP = DELAY(2, W1)
```

Erstellt eine neue Spalte TEMP, in der jeder Wert den Inhalt von Spalte W1 darstellt, der um zwei Zeitschritte verzögert ist.

<p>TEMP = DELAY(10, W1:W3)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind der um zehn Zeitschritte verzögerte Inhalt von Spalte W1, die Werte von Spalte WX sind der um zehn Zeitschritte verzögerte Inhalt von Spalte W2 und die Werte von Spalte WY sind der um zehn Zeitschritte verzögerte Inhalt von Spalte W3.</p>
<p>TEMP = DELAY(1, W1[10:20])</p> <p>Erstellt eine neue Spalte TEMP, deren erste 10 Zellen die Werte in Spalte W1 enthalten, die um einen Zeitschritt verzögert sind (d. h. Zeile 11-20 von Spalte W1). Die anderen Zellen in TEMP sind leer.</p>
<p>TEMP = DELAY(2, W1[1:5]:W2)</p> <p>Erstellt zwei neue Spalten TEMP und WX, die jeweils Werte in Zeile 1-3 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind jeweils die um zwei Zeitschritte verzögerten Zeilen von Spalte W1, die Werte in der Spalte WX sind jeweils die um zwei Zeitschritte verzögerten Zeilen von Spalte W2 (d. h. Zeile 3-5 von Spalte W1 und W2).</p>

Zugehörige Funktionen

Funktion	Beschreibung
LAG	Gibt die Verzögerung der Eingabespaltenwerte um eine angegebene Anzahl von Zeitschritten an.
SLIDE_WINDOW	Erstellt ein Muster aus einem angegebenen Fenster und schiebt es nach unten, um das nächste Muster zu erstellen.

DERIVATIVE



Syntax

DERIVATIVE(Daten [, Divisor])

Parameter

Daten

Die numerischen Werte, deren abgeleitetes Produkt berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Divisor

Ein Wert, durch den jeder Wert in Daten dividiert werden soll. Dabei kann es sich um einen konstanten Wert oder um einen Ausdruck handeln, dessen Auswertung eine Konstante ergibt.

Beschreibung

DERIVATIVE berechnet das abgeleitete Produkt der Werte in einer Zeitreihe. Jeder Wert ist die Differenz zwischen dem aktuellen Wert und dem Wert aus dem nächsten Zeitschritt. Wenn für Divisor ein Wert angegeben ist, wird jeder Wert durch den angegebenen Wert dividiert. DERIVATIVE gibt für jede Eingabespalte eine neue Spalte zurück, die das abgeleitete Produkt der Werte in der entsprechenden Eingabespalte enthält.

Anmerkung: Die Länge der zurückgegebenen Spalte ist eins kleiner als die Länge der Quelldatenspalte (Daten).

Beispiele

TEMP = DERIVATIVE(5)
Erstellt eine neue Spalte TEMP, die nur leere Zellen enthält (es sind mindestens zwei Zellenwerte erforderlich, um ein Ergebnis zu generieren).
TEMP = DERIVATIVE(COLUMN(1, 2, 5))
Erstellt eine neue Spalte TEMP, die die Werte 1 und 3 enthält.
TEMP = DERIVATIVE(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert das abgeleitete Produkt des Inhalts von Spalte W1 darstellt.
TEMP = DERIVATIVE(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils das abgeleitete Produkt des Inhalts von Spalte W1, die Werte von Spalte WX sind das abgeleitete Produkt des Inhalts von Spalte W2 und die Werte von Spalte WY sind das abgeleitete Produkt des Inhalts von Spalte W3.
TEMP = DERIVATIVE(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die Zellen in Zeile 10-20 das abgeleitete Produkt der entsprechenden Zeilen von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = DERIVATIVE(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind das abgeleitete Produkt der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind das abgeleitete Produkt der entsprechenden Zeilen von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
INTEGRAL	Berechnet das Integral der Werte im angegebenen Datenbereich.

DISTANCE

Syntax

DISTANCE(l at1, l ong1, l at2, l ong2[, UNIT_OF_MEASURE] [, PRECISION])

Parameter

lat1

Der Breitengrad des ersten Punkts als Dezimalwert.

long1

Der Längengrad des ersten Punkts als Dezimalwert.

lat2

Der Breitengrad des zweiten Punkts als Dezimalwert.

long2

Der Längengrad des zweiten Punkts als Dezimalwert.

UNIT_OF_MEASURE

Ein optionaler Parameter, der die Maßeinheit für den zurückgegebenen Abstand angibt. Mögliche Werte sind MILES (Meilen) oder KILOMETERS (Kilometer). Wenn Sie diesen Parameter nicht angeben, wird als Standardeinstellung MILES verwendet.

PRECISION

Ein optionaler Parameter, der die Genauigkeit nach dem Dezimalzeichen für den zurückgegebenen Abstand angibt. Wenn Sie einen Genauigkeitswert festlegen, wird der zurückgegebene Abstand auf die angegebene Anzahl von Dezimalstellen gekürzt. Der Maximalwert ist 5. Wenn Sie hierfür keinen Wert angeben, wird die Anzahl der Dezimalstellen nicht gekürzt.

Beschreibung

DISTANCE berechnet den Abstand zwischen zwei Punkten. Breitengrad und Längengrad werden in Dezimalzahleinheiten erwartet.

Beispiele

DISTANCE (18.529747,73.839798,18.533511,73.8777995,MILES,2) gibt einen Wert von 2,50 Meilen zurück.

DISTINCT



Syntax

DISTINCT(Daten)

Parameter

Daten

Der Datenbereich.

Beschreibung

DISTINCT findet die eindeutigen Werte im angegebenen Datenbereich. DISTINCT gibt diese Werteliste in einer einzelnen Spalte zurück. DISTINCT ignoriert beim Vergleich von Zeichenfolgen die Groß-/Kleinschreibung. So werden etwa \042Aaa\042 und \042aAa\042 und \042AAA\042 alle als derselbe Wert gezählt.

Die Makrofunktion DISTINCT benötigt unter Umständen lange Zeit für die Berechnung, wenn die Daten umfangreich sind. Es wird ein Fortschrittsleiste \042Berechnung... \042 angezeigt, bis die Berechnung abgeschlossen ist. Wenn Sie die Berechnung abbrechen möchten, klicken Sie auf das \042X\042 des Fortschrittsleistes und löschen Sie die Funktionsdefinition, die die Makrofunktion DISTINCT enthält.

DIV

Syntax

Daten DIV Divisor; Daten / Divisor

Parameter

Daten

Die numerischen Werte, die dividiert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Divisor

Der oder die Wert(e), durch die die Werte im angegebenen Datenbereich dividiert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Daten2 muss mit der Anzahl der Spalten in Daten1 übereinstimmen, es sei denn, bei Daten2 handelt es sich um eine Konstante. Informationen zur Formatdefinition von Divisor (identisch mit Daten) finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

DIV dividiert den angegebenen Datenbereich durch den Wert des Divisors. DIV gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten durch die entsprechende Spalte von Divisor dividiert wird (d. h., die erste Spalte von Daten wird durch die erste Spalte von Divisor dividiert, die zweite Spalte durch die zweite Spalte usw.).

Wenn es sich bei Divisor um eine Konstante handelt, wird jeder Wert in Daten durch den betreffenden Wert dividiert. Wenn Divisor eine oder mehrere Spalten enthält, werden die Berechnungen zeilenweise zwischen einer Spalte von Daten und einer Spalte von Divisor ausgeführt. Die erste Zeile von Daten wird durch die erste Zeile von Divisor dividiert, die zweite Zeile durch die zweite Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Wenn eine Spalte in jeder Zeile dieselbe Zahl x wie Divisor enthält, ist dies dasselbe, als wenn als Divisor die Konstante x verwendet wird.

Anmerkung: Der Operator DIV kann durch einen Schrägstrich (/) abgekürzt werden.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = 8 DIV 4 oder TEMP = 8/4
Erstellt eine neue Spalte TEMP, die den Wert zwei enthält.
TEMP = W1/8
Erstellt eine neue Spalte TEMP, in der jeder Wert den Inhalt von Spalte W1 darstellt, der durch acht dividiert ist.
TEMP = W1:W3/2
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind der durch zwei dividierte Inhalt von Spalte W1, die Werte von Spalte WX sind der durch zwei dividierte Inhalt von Spalte W2 und die Werte von Spalte WY sind der durch zwei dividierte Inhalt von Spalte W3.
TEMP = W1/W1
Erstellt eine neue Spalte TEMP, die nur Einsen enthält (jeder Wert, der durch sich selbst dividiert wird, ergibt eins).
TEMP = W1/W2
Erstellt eine neue Spalte TEMP, wobei jeder Zeilenwert von Spalte W1 durch den entsprechenden Zeilenwert von Spalte W2 dividiert ist.
TEMP = W1:W3/W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Werte von Spalte W1 durch die entsprechenden Zeilenwerte von Spalte W4 dividiert. Die Spalte WX enthält die Division von Spalte W2 durch Spalte W5. Die Spalte WY enthält die Division von Spalte W3 durch Spalte W6.
TEMP = W1[10:20] / W2 oder TEMP = W1[10:20] / W2[1:11]
Erstellt eine neue Spalte TEMP, deren erste 11 Zellen das Ergebnis der Division der Werte in Zeile 10-20 von Spalte W1 durch die Werte in Zeile 1-11 von Spalte W2 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
MOD	Berechnet den Modulo-Wert des Inhalts des angegebenen Datenbereichs.
MULT	Multipliziert die Inhalte zweier Datenbereiche.
POW	Erhebt einen Basiswert in die Potenz des angegebenen Exponenten.

Syntax

Daten1 EQ Daten2; Daten1 == Daten2; (Daten1 = Daten2)

Parameter

Daten1

Der Zellenbereich, der verglichen werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Die Zahlen, mit denen alle Werte in der angegebenen Spalte verglichen werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Daten2 muss mit der Anzahl der Spalten in Daten1 übereinstimmen, es sei denn, bei Daten2 handelt es sich um eine Konstante. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

EQ vergleicht die beiden angegebenen Datenbereiche und gibt eine Eins zurück, wenn die Werte gleich sind, bzw. eine Null, wenn die Werte ungleich sind. EQ gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten1 mit der entsprechenden Spalte von Daten2 verglichen wird (d. h., die erste Spalte von Daten1 wird mit der ersten Spalte von Daten2 verglichen, die zweite Spalte mit der zweiten Spalte usw.).

Wenn es sich bei Daten2 um eine Konstante handelt, wird jeder Wert in Daten1 mit dem betreffenden Wert verglichen. Wenn es sich bei Daten2 um eine Spalte handelt, werden die Berechnungen zeilenweise ausgeführt. Die erste Zeile von Daten1 wird mit der ersten Zeile von Daten2 verglichen, die zweite Zeile mit der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Beim Vergleich von Zeichenfolgen spielt die Groß-/Kleinschreibung keine Rolle (d. h., "Ja", "ja", "JA" und "jA" werden gleich behandelt).

Anmerkung: Der Operator EQ kann durch ein doppeltes Gleichheitszeichen (==) abgekürzt werden. Innerhalb runder Klammern kann auch ein einfaches Gleichheitszeichen (=) für die Makrofunktion EQ verwendet werden (außerhalb runder Klammern wird das Gleichheitszeichen als Zuweisungsoperator interpretiert).

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = 3 EQ 4 oder TEMP = 3==4 oder TEMP = (3=4)
Erstellt eine neue Spalte TEMP, die den Wert null enthält (drei ist ungleich vier).
TEMP = "Ja" == "JA"
Erstellt eine neue Spalte TEMP, die den Wert eins enthält (beim Vergleich von Zeichenfolgen wird die Groß-/Kleinschreibung nicht beachtet).
TEMP = W1 == 8
Erstellt eine neue Spalte TEMP, in der jeder Wert eins ist, wenn der entsprechende Zeilenwert von Spalte W1 gleich der Zahl acht ist; andernfalls ist der Wert null.
TEMP = W1==W1
Erstellt eine neue Spalte TEMP, die nur Einsen enthält (jede Zahl ist gleich sich selbst).
TEMP = W1==W2
Erstellt eine neue Spalte TEMP, wobei jeder Zeilenwert von Spalte W1 mit dem entsprechenden Zeilenwert von Spalte W2 verglichen wird.
TEMP = W1:W3 == W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Werte von Spalte W1 mit den entsprechenden Zeilenwerten von Spalte W4 verglichen. Die Spalte WX enthält den Vergleich von Spalte W2 mit Spalte W5. Die Spalte WY enthält den Vergleich von Spalte W3 mit Spalte W6.
TEMP = W1[10:20] == W2 oder TEMP = W1[10:20] == W2[1:11]
Erstellt eine neue Spalte TEMP, deren erste 11 Zellen das Ergebnis des Vergleichs der Werte in Zeile 10-20 von Spalte W1 mit den Werten in Zeile 1-11 von Spalte W2 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
EQ	Gibt TRUE zurück, wenn der eine Datenbereich dem anderen gleich ist.
GE	Gibt TRUE zurück, wenn der eine Datenbereich größer-gleich dem anderen ist.
GT	Gibt TRUE zurück, wenn der eine Datenbereich größer als der andere ist.
LE	Gibt TRUE zurück, wenn der eine Datenbereich kleiner-gleich dem anderen ist.
LT	Gibt TRUE zurück, wenn der eine Datenbereich kleiner als der andere ist.
NE	Gibt TRUE zurück, wenn der eine Datenbereich dem anderen ungleich ist.

EXP

Syntax

EXP(Daten)

Parameter

Daten

Die numerischen Werte, die als Exponenten für die natürliche Zahl e verwendet werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

EXP erhebt die natürliche Zahl, e , in die Potenz, die jeweils durch den Wert im angegebenen Datenbereich angegeben wird (berechnet wird also e^x). Die Konstante e hat den Wert 2,7182818. EXP gibt für jede Eingabespalte eine neue Spalte zurück, die das Ergebnis e^x für jeden Wert x in der entsprechenden Eingabespalte enthält. EXP ist die Umkehrfunktion der Makrofunktion LN.

Anmerkung: Wenn der Wert x zu groß oder zu klein ist, wird ein Überlauffehler zurückgegeben. Dies ist der Fall, wenn e^x den maximalen oder minimalen 32-Bit-Gleitkommawert überschreitet.

Beispiele

TEMP = EXP(2)	Erstellt eine neue Spalte TEMP, die den Wert 7,39 enthält.
TEMP = EXP(W1)	Erstellt eine neue Spalte TEMP, in der jeweils e in die Potenz erhoben wird, die durch den Inhalt von Spalte W1 angegeben wird.
TEMP = EXP(W1:W3)	Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP ergeben sich durch Potenzierung von e um die in Spalte W1 angegebenen Werte, die Werte von Spalte WX ergeben sich durch Potenzierung von e um die in Spalte W2 angegebenen Werte und die Werte von Spalte WY ergeben sich durch Potenzierung von e um die in Spalte W3 angegebenen Werte.
TEMP = EXP(W1[10:20])	Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils die Potenzierung von e um die Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = EXP(W1[1:5]:W2)	Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP stellen die Potenzierung von e um die entsprechenden Zeilenwerte von Spalte W1 dar, die Werte in der Spalte WX stellen die Potenzierung von e um die entsprechenden Zeilenwerte von Spalte W2 dar.

Zugehörige Funktionen

Funktion	Beschreibung
LN oder LOG	Berechnet den natürlichen Logarithmus des Inhalts des angegebenen Datenbereichs.

Funktion	Beschreibung
LOG2	Berechnet den binären Logarithmus des Inhalts des angegebenen Datenbereichs.
LOG10	Berechnet den dekadischen Logarithmus des Inhalts des angegebenen Datenbereichs.
POW	Erhebt einen Basiswert in die Potenz des angegebenen Exponenten.

EXTERNALCALLOUT



Syntax

EXTERNALCALLOUT(*Aufrufname*, *Arg1*, ...)

Parameter

Aufrufname

Der Name des mithilfe der ExternalCallout-API erstellten Aufrufs. Dieser Name muss mit dem Namen der ExternalCallout-Kategorie übereinstimmen, die Sie in IBM Marketing Platform erstellt haben.

Arg1

Ein für Ihren Aufruf gegebenenfalls erforderliches Argument.

Beschreibung

Mit EXTERNALCALLOUT können Sie eine externe Anwendung aufrufen, um Ihrem interaktiven Ablaufdiagramm Daten hinzuzufügen. EXTERNALCALLOUT kann alle Daten zurückgeben, die von Ihrem Aufruf zurückgegeben werden. Sie müssen diesen Aufruf mithilfe der ExternalCallout-API in Java schreiben. Weitere Informationen finden Sie im *IBM Interact-Entwicklerhandbuch*.

Beispiele

EXTERNALCALLOUT(getStockPrice, UNCA)

Ruft den Aufruf getStockPrice auf und übergibt den Namen der Aktie, UNCA, als Argument. Dieser benutzerdefinierte Aufruf gibt den Aktienkurs zurück, wie er durch den Aufruf definiert ist.

EXTRACT



Syntax

EXTRACT(Prädikatspalte, Daten)

Parameter

Prädikatspalte

Eine Spalte mit booleschen Werten oder ein Ausdruck, dessen Auswertung eine einzelne Spalte mit booleschen Werten ergibt. Boolesche Werte werden als null oder ungleich null interpretiert. Diese Spalte sollte mindestens so viele Zeilen enthalten wie der Datenbereich, aus dem die Daten extrahiert werden. Andernfalls wird Prädikatspalte auf die Anzahl der Zeilen begrenzt, die von der Makrofunktion EXTRACT verarbeitet werden (siehe die folgende "Beschreibung").

Daten

Die Werte, die extrahiert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

EXTRACT gibt die Zeilen im angegebenen Datenbereich zurück, die in der entsprechenden Zeile der Prädikatspalte den Wert eins aufweisen. Diese Makrofunktion reduziert Daten, indem sie alle Zeilen aussondert, deren entsprechender Wert in Prädikatspalte null ist. EXTRACT gibt für jede Eingabespalte eine neue Spalte zurück, die die Werte in der jeweiligen Eingabespalte enthält, deren entsprechender Wert in Prädikatspalte eins ist. Die extrahierten Zeilen von Daten belegen die ersten n Zellen der Ausgabespalten, wobei n die Anzahl der Einsen in Prädikatspalte ist.

Da EXTRACT zeilenweise vorgeht, wird für jede Zeile bis zum letzten Wert der kürzesten Spalte (d. h. der kürzesten Spalte von Prädikatspalte und den Spalten im Datenbereich Daten) ein Ergebnis erzeugt. Alle Werte ungleich null in Prädikatspalte ergeben in der Auswertung eins.

Anmerkung: In der Regel empfiehlt es sich, eine Prädikatspalte zu erstellen, die eine der Vergleichsmakrofunktionen verwendet (`==`, `>`, `<`, `ISEVEN`, `ISODD` usw.). Dann können die fraglichen Zeilen mithilfe der Makrofunktion EXTRACT aus einem angegebenen Datenbereich extrahiert werden. Dies kann dazu verwendet werden, ungültige Dateneinträge auszusondern (etwa wenn ein bestimmter Wert den Maximal- oder Minimalwert für eine Datenvariable überschreitet). Darüber hinaus können alle Beispiele einer bestimmten Klasse konsolidiert werden (wenn z. B. die Spalte W3 bei einer der Ausgabeklassen Einsen und Nullen enthält, können Sie mit `W4=EXTRACT(W3, W1:W2)` die Eingaben W1 und W2 extrahieren). Da EXTRACT alle extrahierten Zeilen zu einem Datenblock komprimiert (d. h., den Zellenbereich `WX[1:n]:WY` füllt, wobei n die Anzahl der extrahierten Zeilen ist), kann die Funktion dazu verwendet werden, einen Zellenbereich von den aktuellen Zeilenpositionen in Zeile 1 bis n des Arbeitsblatts zu kopieren.

Beispiele

<code>TEMP = EXTRACT(1, W1)</code> Erstellt eine neue Spalte TEMP, die eine Kopie von Spalte W1 enthält.
--

TEMP = EXTRACT(1, W1[50:100]:W2)
Erstellt zwei neue Spalten TEMP und WX, die in den ersten 51 Zellen Werte enthalten. Die Werte in der Spalte TEMP sind die Zellen 50-100 von Spalte W1, die Werte in der Spalte WX sind die Zellen 50-100 von Spalte W2.
TEMP = EXTRACT(W3, W1:W2)
Erstellt zwei neue Spalten TEMP und WX. Bei jeder Zeile, bei der der Wert in Spalte W3 eins ist, wird die entsprechende Zeile in den Spalten W1 und W2 in Spalte TEMP bzw. WX extrahiert.
TEMP = EXTRACT(W1>W2, W1)
Erstellt eine neue Spalte TEMP, die alle Werte in Spalte W1 enthält, die größer als die entsprechenden Werte in Spalte W2 sind.
TEMP = EXTRACT(W3[10:20], W1[10:20]:W2)
Erstellt zwei neue Spalten TEMP und WX. Bei den Zeilen 10-20, bei denen der Wert in Spalte W3 eins ist, wird die entsprechende Zeile in den Spalten W1 und W2 in Spalte TEMP bzw. WX extrahiert.

Zugehörige Funktionen

Funktion	Beschreibung
IF	Leitet eine If-then-else-Anweisung einer bedingten Aussage ein.
SELECT	Gibt die angegebene(n) Spalte(n) aus einem Datenbereich zurück.
SUBSAMPLE	Reduziert Daten durch Rückgabe jedes n-ten Zeilenwerts.
SUBSTITUTE	Ersetzt Werte in einer Spalte durch einen Wert, der in einer Konvertierungstabelle angegeben ist.

FACTORIAL

Syntax

FACTORIAL(Daten)

Parameter

Daten

Die Ganzzahlwerte, deren Fakultät berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt; der Wert muss jedoch größer-gleich null sein. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

FACTORIAL berechnet die Fakultät der Werte im angegebenen Datenbereich. Alle Eingabewerte müssen Ganzzahlen größer-gleich null sein. Die Fakultät einer Ganz-

zahl kleiner-gleich eins ist eins. Bei Ganzzahlen $x \geq 2$ ist die Fakultät $x! = x(x-1)(x-2)\dots(x-(x-1))$. FACTORIAL gibt für jede Eingabespalte eine neue Spalte zurück, die die Fakultät der Zahlen in der entsprechenden Eingabespalte enthält.

Anmerkung: Jeder Wert größer als 34 erzeugt die Ausgabe ??? (Gleitkomma-Überlauffehler).

Beispiele

TEMP = FACTORIAL(3)
Erstellt eine neue Spalte TEMP, die den Wert 6 enthält.
TEMP = FACTORIAL(-2)
Generiert einen Fehler 333, der darauf hinweist, dass das Argument größer-gleich 0 sein muss.
TEMP = FACTORIAL(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert die Fakultät des Inhalts von Spalte W1 darstellt.
TEMP = FACTORIAL(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils die Fakultät des Inhalts von Spalte W1, die Werte von Spalte WX sind die Fakultät des Inhalts von Spalte W2 und die Werte von Spalte WY sind die Fakultät des Inhalts von Spalte W3.
TEMP = FACTORIAL(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils die Fakultät der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = FACTORIAL(W1[50:99]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils Werte in Zeile 1-50 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind jeweils die Fakultät der Zeilen von Spalte W1, die Werte in der Spalte WX sind die Fakultät der Zeilen von Spalte W2.

FLOOR

Syntax

FLOOR(Daten)

Parameter

Daten

Die numerischen Werte, deren Untergrenze berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

FLOOR berechnet die Untergrenze der Werte im angegebenen Datenbereich. Die Untergrenze einer Zahl ist die größte Ganzzahl, die kleiner als die Zahl ist. FLOOR gibt

für jede Eingabespalte eine neue Spalte zurück, die die Untergrenze der Zahlen in der entsprechenden Eingabespalte enthält.

Anmerkung: FLOOR ist mit der Makrofunktion INT identisch.

Beispiele

TEMP = FLOOR(4,3)
Erstellt eine neue Spalte TEMP, die den Wert 4 enthält.
TEMP = FLOOR(-2,9)
Erstellt eine neue Spalte TEMP, die den Wert -3 enthält.
TEMP = FLOOR(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert die Untergrenze des Inhalts von Spalte W1 darstellt.
TEMP = FLOOR(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils die Untergrenze des Inhalts von Spalte W1, die Werte von Spalte WX sind die Untergrenze des Inhalts von Spalte W2 und die Werte von Spalte WY sind die Untergrenze des Inhalts von Spalte W3.
TEMP = FLOOR(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils die Untergrenze der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = FLOOR(W1[50:99]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils Werte in Zeile 1-50 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind jeweils die Untergrenze der Zeilen von Spalte W1, die Werte in der Spalte WX sind die Untergrenze der Zeilen von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
CEILING	Berechnet die Obergrenze jedes Werts im angegebenen Datenbereich.
FRACTION	Berechnet die Nachkommastellen jedes Werts im angegebenen Datenbereich.
TRUNCATE	Berechnet den Ganzzahlanteil jedes Werts im angegebenen Datenbereich.

FORMAT

Syntax

Format weist zwei Formen auf, eine für numerische Datentypen und eine für Text-/Zeichendatentypen.

Bei numerischen Datentypen:

FORMAT(Spaltenname, Breite [, Genauigkeit [, Formattyp [, Ausrichtung [, Auffüllung]]]])

Bei Text-/Zeichendatentypen:

FORMAT(Spaltenname, Breite [, Ausrichtung])

Parameter

Spaltenname

Das Makro untersucht Spaltenname und ermittelt dessen Datentyp, um dementsprechend die jeweiligen Regeln für nachfolgende Parameter anzuwenden.

Breite

Die Breite muss groß genug sein, um das vollständige Ergebnis aufnehmen zu können; andernfalls wird das Ergebnis abgeschnitten. Die zulässigen Werte reichen von 1 bis 29, wenn Spaltenname numerisch ist, andernfalls von 1 bis 255.

Genauigkeit

Die Genauigkeit ist die Anzahl der Stellen nach dem Dezimalzeichen. Die zulässigen Werte reichen von 0 bis 15. Wenn sie null ist, ist das Ergebnis eine Ganzzahl. Der Standardwert für die Genauigkeit ist 2.

Formattyp

Gültige Schlüsselwörter für Formattyp:

PERIOD	Ein Punkt (.) wird als Dezimalzeichen verwendet. Es wird kein Trennzeichen verwendet. Dies ist der Standardwert.
COMMA	Als Dezimalzeichen wird ein Komma (,) verwendet. Es wird kein Trennzeichen verwendet.
PERIOD_COMMA	Als Dezimalzeichen wird ein Punkt und als Trennzeichen ein Komma verwendet.
COMMA_PERIOD	Als Dezimalzeichen wird ein Komma und als Trennzeichen ein Punkt verwendet.

Ausrichtung

Gültige Schlüsselwörter für Ausrichtung sind LEFT und RIGHT. Bei numerischen Datentypen ist der Standardwert RIGHT, bei Text-/Zeichendatentypen LEFT.

Auffüllung

Gültige Schlüsselwörter für Auffüllung sind SPACE und ZERO. Der Standardwert ist SPACE. Wenn Ausrichtung den Wert LEFT aufweist, wird ZERO ignoriert (und stattdessen SPACE verwendet).

Numerische Zeichenfolgen innerhalb eines Text-/Zeichendatentyps werden als Text/Zeichen behandelt. Außerdem unterstützt die numerische Form mehrere optionale Schlüsselwörter, die jeweils einen Standardwert aufweisen. Um jedoch den Standardwert des zweiten oder eines nachfolgenden optionalen Schlüsselworts zu überschreiben, MÜSSEN Sie die Standardwerte für die vorhergehenden optionalen Schlüsselwörter angeben (sie werden somit obligatorisch). Um zum Beispiel den

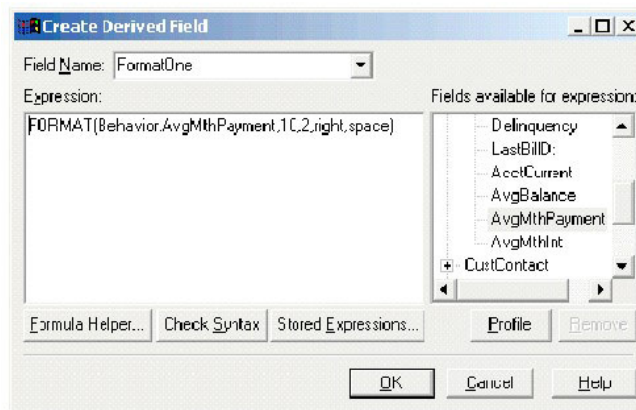
Standardwert für Ausrichtung durch LEFT zu überschreiben, müssen Sie Folgendes codieren: FORMAT(meineZahlenspalte, 10, 2, PERIOD, LEFT).

Beschreibung

FORMAT konvertiert numerische Daten in eine Zeichenfolgeform, wobei die Ausgabezeichenfolge durch verschiedene Formatierungsoptionen gesteuert und definiert werden kann. Dies ist besonders nützlich, wenn Momentaufnahmedateien mit bestimmten Formaten für das Mailing von Dateien erstellt werden sollen.

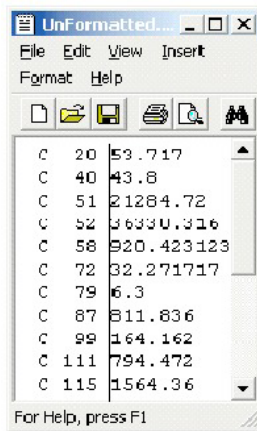
Beispiele

Das folgende Beispiel definiert mithilfe von FORMAT ein abgeleitetes Feld.



Die folgenden Beispiele zeigen dasselbe Feld, AvgMthPayment, in drei Formaten.

Unformatiert:



Mit `FORMAT(Behavior.AvgMthPayment,10,2,right.space)` formatiert:

Label	Value	Formatted Value
f	20	53.72
C	40	43.80
C	51	21284.72
C	52	36330.32
C	58	920.42
C	72	32.27
C	79	6.30
C	87	811.84
C	99	164.16
C	111	794.47
C	115	1564.36

Mit `FORMAT(Behavior.AvgMthPayment,10,4)` formatiert:

Label	Value	Formatted Value
f	20	53.7170
C	40	43.8000
C	51	21284.7200
C	52	36330.3160
C	58	920.4231
C	72	32.2717
C	79	6.3000
C	87	811.8360
C	99	164.1620
C	111	794.4720
C	115	1564.3600

FRACTION

Syntax

`FRACTION(Daten)`

Parameter

Daten

Die numerischen Werte, deren Nachkommastellen berechnet werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

`FRACTION` berechnet die Nachkommastellen der Werte im angegebenen Datenbereich. Dabei wird für jede Eingabespalte eine neue Spalte zurückgegeben, die die Nachkommastellen der Zahlen in der entsprechenden Eingabespalte enthält.

Anmerkung: Die Makrofunktionen FRACTION und TRUNCATE sind komplementär, da sie in der Summe die ursprünglichen Werte ergeben.

Beispiele

TEMP = FRACTION(4,3)
Erstellt eine neue Spalte TEMP, die den Wert 0,3 enthält.
TEMP = FRACTION(-2,9)
Erstellt eine neue Spalte TEMP, die den Wert -0,9 enthält.
TEMP = FRACTION(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert die Nachkommastellen des Inhalts von Spalte W1 darstellt.
TEMP = FRACTION(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils die Nachkommastellen des Inhalts von Spalte W1, die Werte von Spalte WX sind die Nachkommastellen des Inhalts von Spalte W2 und die Werte von Spalte WY sind die Nachkommastellen des Inhalts von Spalte W3.
TEMP = FRACTION(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils die Nachkommastellen der Werte in Zeile 10-20 von Spalte W1 enthalten. Die anderen Zellen in TEMP sind leer.
TEMP = FRACTION(W1[50:99]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils Werte in Zeile 1-50 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind jeweils die Nachkommastellen der Zeilen von Spalte W1, die Werte in der Spalte WX sind die Nachkommastellen der Zeilen von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
CEILING	Berechnet die Obergrenze jedes Werts im angegebenen Datenbereich.
FLOOR	Berechnet die Untergrenze jedes Werts im angegebenen Datenbereich.
TRUNCATE	Berechnet den Ganzzahlanteil jedes Werts im angegebenen Datenbereich.

GAUSS



Syntax

GAUSS(Daten [, Mittelwert, Stdabw])

Parameter

Daten1

Der Zellenbereich, dessen gaußscher Wert berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Mittelwert

Der Mittelwert der gaußschen Kurve. Wird dieser Parameter nicht angegeben, ist der Standard null. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Mittelwert muss mit der Anzahl der Spalten in Daten übereinstimmen, es sei denn, bei Mittelwert handelt es sich um eine Konstante oder eine einzelne Spalte. Informationen zur Formatdefinition von Mittelwert (identisch mit der Definition von Daten) finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Stdabw

Die Standardabweichung der gaußschen Kurve. Wird dieser Parameter nicht angegeben, ist der Standard eins. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Stdabw muss mit der Anzahl der Spalten in Daten übereinstimmen, es sei denn, bei Stdabw handelt es sich um eine Konstante oder eine einzelne Spalte. Informationen zur Formatdefinition von Stdabw (identisch mit der Definition von Daten) finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

GAUSS berechnet den gaußschen Wert der Zahlen im angegebenen Datenbereich. Dabei wird für jede Eingabespalte eine neue Spalte zurückgegeben, die den gaußschen Wert der entsprechenden Eingabe enthält. GAUSS wird wie folgt berechnet:

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

Die Parameter für Mittelwert und Standardabweichung werden wie folgt behandelt:

- Wenn kein Mittelwert und keine Standardabweichung angegeben werden, wird null bzw. eins verwendet.
- Wenn Mittelwert und Stdabw Konstanten sind, werden diese Werte dazu verwendet, den gaußschen Wert für alle Werte in Daten anzugeben.
- Wenn Mittelwert und Stdabw einzelne Spalten sind, werden die entsprechenden Zeilenwerte für jede Zeile von Daten verwendet.
- Wenn Mittelwert und Stdabw Spaltenbereiche sind (beide müssen dieselbe Anzahl von Spalten wie Daten aufweisen), verwendet jede Zelle in Daten ihr jeweiliges Paar von entsprechenden Zellen in Mittelwert und Stdabw.

Beispiele

<code>TEMP = GAUSS(0)</code> oder <code>TEMP = GAUSS(0, 0, 1)</code>
Erstellt eine neue Spalte TEMP, die den Wert 0,4 enthält.
<code>TEMP = GAUSS(W1)</code>
Erstellt eine neue Spalte TEMP, in der jeder Wert den gaußschen Wert der entsprechenden Zeile von Spalte W1 darstellt (bei Mittelwert null und Einheitsvarianz).
<code>TEMP = GAUSS(W1:W3)</code>
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die gaußschen Werte von Spalte W1, die Werte von Spalte WX sind die gaußschen Werte von Spalte W2 und die Werte von Spalte WY sind die gaußschen Werte von Spalte W3. Der gaußsche Wert verwendet Mittelwert null und Einheitsvarianz.
<code>TEMP = GAUSS(W1[1:50]:W3)</code>
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die gaußschen Werte von Zeile 1-50 von Spalte W1, die Werte von Spalte WX sind die gaußschen Werte von Zeile 1-50 von Spalte W2 und die Werte von Spalte WY sind die gaußschen Werte von Zeile 1-50 von Spalte W3. Der gaußsche Wert verwendet Mittelwert null und Einheitsvarianz.
<code>TEMP = GAUSS(W1, 0, 3,5)</code>
Erstellt eine neue Spalte TEMP, in der jeder Wert den gaußschen Wert der entsprechenden Zeile von Spalte W1 darstellt. Die gaußsche Kurve hat den Mittelwert 0 und die Standardabweichung 3,5.
<code>TEMP = GAUSS(W1, W2, W3)</code>
Erstellt eine neue Spalte TEMP, wobei jeder Wert den gaußschen Wert von W1 darstellt, den den entsprechenden Zeilenwert von Spalte W2 als Mittelwert und den entsprechenden Zeilenwert von Spalte W3 als Standardabweichung verwendet.
<code>TEMP = GAUSS(W1:W2, W3:W4, W5:W6)</code>
Erstellt zwei neue Spalten TEMP und WX. Die Spalte TEMP enthält die gaußschen Werte der Werte in Spalte W1, wobei die entsprechenden Zeilen von Spalte W3 als Mittelwert und die entsprechenden Zeilen von Spalte W5 als Standardabweichung verwendet werden. Die Spalte WX enthält die gaußschen Werte der Werte in Spalte W2, wobei die entsprechenden Zeilen von Spalte W4 als Mittelwert und die entsprechenden Zeilen von Spalte W6 als Standardabweichung verwendet werden.

Zugehörige Funktionen

<code>RANDOM_GAUSS</code>	Gibt die angegebene Anzahl von Zufallswerten aus einer gaußschen Verteilung zurück.
---------------------------	---

GAUSS_AREA



Syntax

`GAUSS_AREA(Daten [, Mittelwert, Stdabw])`

Parameter

Daten1

Der Zellenbereich, für den die Fläche unter der gaußschen Kurve berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Mittelwert (identisch mit der Definition von Daten) finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Mittelwert

Der Mittelwert der gaußschen Kurve. Wird dieser Parameter nicht angegeben, ist der Standard null. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Mittelwert muss mit der Anzahl der Spalten in Daten übereinstimmen, es sei denn, bei Mittelwert handelt es sich um eine Konstante oder eine einzelne Spalte. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

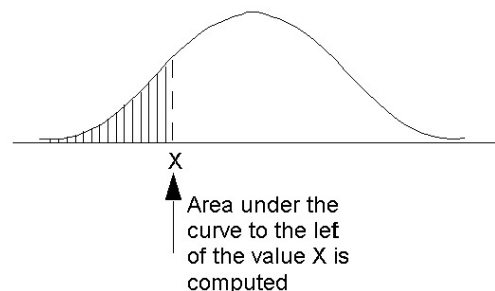
Stdabw

Die Standardabweichung der gaußschen Kurve. Wird dieser Parameter nicht angegeben, ist der Standard eins. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Stdabw muss mit der Anzahl der Spalten in Daten übereinstimmen, es sei denn, bei Stdabw handelt es sich um eine Konstante oder eine einzelne Spalte. Informationen zur Formatdefinition von Stdabw (identisch mit der Definition von Daten) finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

GAUSS_AREA berechnet die Fläche unter der gaußschen Kurve (von $-\infty$ bis zum angegebenen Datenwert X) der Zahlen im angegebenen Datenbereich (siehe die folgende Abbildung). Dabei wird für jede Eingabespalte eine neue Spalte zurückgegeben, die die Fläche unter der gaußschen Kurve der entsprechenden Eingabe enthält.

Area Under a Gaussian Computed by GAUSS_AREA



Die Parameter für Mittelwert und Standardabweichung werden wie folgt behandelt:

- Wenn kein Mittelwert und keine Standardabweichung angegeben werden, wird null bzw. eins verwendet.
- Wenn Mittelwert und Stdabw Konstanten sind, werden diese Werte dazu verwendet, den gaußschen Wert für alle Werte in Daten anzugeben.
- Wenn Mittelwert und Stdabw einzelne Spalten sind, werden die entsprechenden Zeilenwerte für jede Zeile von Daten verwendet.
- Wenn Mittelwert und Stdabw Spaltenbereiche sind (beide müssen dieselbe Anzahl von Spalten wie Daten aufweisen), verwendet jede Zelle in Daten ihr jeweiliges Paar von entsprechenden Zellen in Mittelwert und Stdabw.

Anmerkung: Im letztgenannten Fall, wenn Mittelwert und Stdabw Spaltenbereiche sind, legt die Länge der jeweiligen Spalte fest, wie viele Zeilen in der entsprechenden Ausgabespalte enthalten sind. Wenn es sich bei einer Spalte von Mittelwert oder Stdabw um eine einzelne Zelle handelt, wird der betreffende Wert für alle Zeilenwerte von Daten verwendet. Wenn Mittelwert oder Stdabw mehrere Zeilen enthält, werden die entsprechenden Zeilen berechnet. Zeilen von Daten, für die in Mittelwert und Stdabw keine entsprechenden Werte vorliegen, werden nicht berechnet.

Beispiele

<p>TEMP = GAUSS_AREA(0) oder TEMP = GAUSS_AREA(0, 0, 1)</p> <p>Erstellt eine neue Spalte TEMP, die den Wert 0,5 enthält.</p>
<p>TEMP = GAUSS_AREA(W1)</p> <p>Erstellt eine neue Spalte TEMP, in der jeder Wert die Fläche unter der gaußschen Kurve der entsprechenden Zeile von Spalte W1 darstellt (bei Mittelwert null und Einheitsvarianz).</p>
<p>TEMP = GAUSS_AREA(W1:W3)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die Flächen unter den gaußschen Kurven von Spalte W1, die Werte von Spalte WX sind die Flächen unter den gaußschen Kurven von Spalte W2 und die Werte von Spalte WY sind die Flächen unter den gaußschen Kurven von Spalte W3. Der gaußsche Wert verwendet Mittelwert null und Einheitsvarianz.</p>
<p>TEMP = GAUSS_AREA(W1[1:50]:W3)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die Flächen unter den gaußschen Kurven von Zeile 1-50 von Spalte W1, die Werte von Spalte WX sind die Flächen unter den gaußschen Kurven von Zeile 1-50 von Spalte W2 und die Werte von Spalte WY sind die Flächen unter den gaußschen Kurven von Zeile 1-50 von Spalte W3. Der gaußsche Wert verwendet Mittelwert null und Einheitsvarianz.</p>
<p>TEMP = GAUSS_AREA(W1, 0, 3,5)</p> <p>Erstellt eine neue Spalte TEMP, in der jeder Wert die Fläche unter der gaußschen Kurve der entsprechenden Zeile von Spalte W1 darstellt. Die gaußsche Kurve hat den Mittelwert 0 und die Standardabweichung 3,5.</p>
<p>TEMP = GAUSS_AREA(W1, W2, W3)</p> <p>Erstellt eine neue Spalte TEMP, wobei jeder Wert die Fläche unter der gaußschen Kurve von W1 darstellt; dabei wird der entsprechende Zeilenwert von Spalte W2 als Mittelwert und der entsprechende Zeilenwert von Spalte W3 als Standardabweichung verwendet.</p>

```
TEMP = GAUSS_AREA(W1:W2, W3:W4, W5:W6)
```

Erstellt zwei neue Spalten TEMP und WX. Die Spalte TEMP enthält die Flächen unter den gaußschen Kurven der Werte in Spalte W1; dabei werden die entsprechenden Zeilen von Spalte W3 als Mittelwert und die entsprechenden Zeilen von Spalte W5 als Standardabweichung verwendet. Die Spalte WX enthält die Flächen unter den gaußschen Kurven der Werte in Spalte W2; dabei werden die entsprechenden Zeilen von Spalte W4 als Mittelwert und die entsprechenden Zeilen von Spalte W6 als Standardabweichung verwendet.

GE

Syntax

Daten1 GE Daten2; Daten1 >= Daten2

Parameter

Daten1

Der numerische Zellenbereich, der verglichen werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Die Zahlen, mit denen alle Werte in der angegebenen Spalte verglichen werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Daten2 muss mit der Anzahl der Spalten in Daten1 übereinstimmen, es sei denn, bei Daten2 handelt es sich um eine Konstante. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

GE vergleicht die beiden angegebenen Datenbereiche und gibt eine Eins zurück, wenn die Werte im ersten Dataset größer-gleich den Werten im zweiten Dataset sind, andernfalls eine Null. GE gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten1 mit der entsprechenden Spalte von Daten2 verglichen wird (d. h., die erste Spalte von Daten1 wird mit der ersten Spalte von Daten2 verglichen, die zweite Spalte mit der zweiten Spalte usw.).

Wenn es sich bei Daten2 um eine Konstante handelt, wird jeder Wert in Daten1 mit dem betreffenden Wert verglichen. Wenn es sich bei Daten2 um eine Spalte handelt, werden die Berechnungen zeilenweise ausgeführt. Die erste Zeile von Daten1 wird mit der ersten Zeile von Daten2 verglichen, die zweite Zeile mit der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Der Operator GE kann durch ein Größer-als-Zeichen, auf das ein Gleichheitszeichen folgt, abgekürzt werden (>=).

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

<code>TEMP = 9 GE 4</code> oder <code>TEMP = 9 >= 4</code>
Erstellt eine neue Spalte TEMP, die den Wert eins enthält (neun ist größer als vier).
<code>TEMP = W1 >= 8</code>
Erstellt eine neue Spalte TEMP, in der jeder Wert eins ist, wenn der entsprechende Zeilenwert von Spalte W1 größer-gleich der Zahl acht ist; andernfalls ist der Wert null.
<code>TEMP = W1:W3 >= 2</code>
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind der mit zwei verglichene Inhalt von Spalte W1, die Werte von Spalte WX sind der mit zwei verglichene Inhalt von Spalte W2 und die Werte von Spalte WY sind der mit zwei verglichene Inhalt von Spalte W3.
<code>TEMP = W1 >= W1</code>
Erstellt eine neue Spalte TEMP, die nur Einsen enthält (jede Zahl ist gleich sich selbst).
<code>TEMP = W1 >= W2</code>
Erstellt eine neue Spalte TEMP, wobei jeder Zeilenwert von Spalte W1 mit dem entsprechenden Zeilenwert von Spalte W2 verglichen wird.
<code>TEMP = W1:W3 >= W4:W6</code>
Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Werte von Spalte W1 mit den entsprechenden Zeilenwerten von Spalte W4 verglichen. Die Spalte WX enthält den Vergleich von Spalte W2 mit Spalte W5. Die Spalte WY enthält den Vergleich von Spalte W3 mit Spalte W6.
<code>TEMP = W1[10:20] >= W2</code> oder <code>TEMP = W1[10:20] >= W2[1:11]</code>
Erstellt eine neue Spalte TEMP, deren erste 11 Zellen das Ergebnis des Vergleichs der Werte in Zeile 10-20 von Spalte W1 mit den Werten in Zeile 1-11 von Spalte W2 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

NE	Gibt TRUE zurück, wenn der eine Datenbereich dem anderen ungleich ist.
----	--

GRID



Syntax

`GRID(Spalte1 [, Spalte2]...)`

Parameter

Spalte1

Die erste Spalte, aus der das Grid erzeugt werden soll. Dabei kann es sich um eine Konstante, eine Spalte, einen einspaltigen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt.

Spalte2

Eine oder mehrere weitere Spalte(n) für die Erzeugung des Grids. Dabei kann es sich um eine Konstante, eine Spalte, einen einspaltigen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt.

Beschreibung

GRID generiert ausgehend von den Zellen in jeder der Eingabespalten ein Grid aller möglichen Wertekombinationen. GRID gibt für jede Eingabespalte eine neue Spalte zurück. Wenn die Eingaben Konstanten enthalten, wird jeder Wert in den anderen Eingabespalten mit dem betreffenden konstanten Wert kombiniert.

Wenn für die erste Zelle in Spalte W1 die Notation W1[1], für die zweite Zelle W1[2] verwendet wird (usw.), erzeugt der Ausdruck GRID(W1, W2) folgendes Ergebnis:

```
W1[1] W2[1] W1[1] W2[2] W1[1] W2[3] : : W1[2] W2[1] W1[2] W2[2] W1[2] W2[3]
: : : :
```

Alle Zellen in Spalte W1 werden paarweise mit jeder Zelle in Spalte W2 verbunden. Bei mehr als zwei Eingabespalten verhält sich GRID entsprechend. Die Anzahl der generierten Zeilen ist gleich dem Produkt der Länge der Eingabespalten.

Anmerkung: Die resultierende Länge aller Eingabeargumente kann nicht größer sein als $(2^{16} - 1)/16$.

Beispiele

TEMP = GRID(1, 2)
Erstellt zwei neue Spalten TEMP und WX, die den Wert 1 bzw. 2 enthalten.
TEMP = GRID(COLUMN(1, 2), COLUMN(3, 4))
Erstellt zwei neue Spalten TEMP und WX. Die Zeilen in diesen beiden Spalten haben folgenden Inhalt: 1 3 1 4 2 3 2 4
TEMP = GRID(W1)
Erstellt eine neue Spalte TEMP, die eine Kopie der Werte in Spalte W1 enthält.
TEMP = GRID(W1, 3)
Erstellt zwei neue Spalten TEMP und WX. Die Spalte TEMP ist eine Kopie von Spalte W1, die Spalte WX enthält für jede Zeile von Spalte W1 den Wert 3.
TEMP = GRID(W1, W2)
Erstellt zwei neue Spalten TEMP und WX, wobei jede Zeile eine der möglichen Zellenkombinationen darstellt (siehe Beschreibung).
TEMP = GRID(W1, W3, W7)
Erstellt drei neue Spalten TEMP, WX und WY, wobei jede Zeile eine der möglichen Zellenkombinationen darstellt (siehe Beschreibung).

Zugehörige Funktionen

Funktion	Beschreibung
SLIDE_WINDOW	Erstellt ein Muster aus einem angegebenen Fenster und schiebt es nach unten, um das nächste Muster zu erstellen.

GROUPBY

Syntax

GROUPBY(Gruppenfeld, Schlüsselwort, Rollup-Feld[, Ausgabefeld])

Parameter

- Gruppenfeld
Gibt die Variable an, nach der die Einträge gruppiert werden (d. h., alle identischen Werte der angegebenen Variable werden derselben Gruppe zugeordnet).
- Schlüsselwort
Gibt die zusammenfassende Rollup-Funktion an, die auf das Rollup-Feld angewendet werden soll.
- Rollup-Feld
Gibt die Variable an, die zusammengefasst werden soll.
- Ausgabefeld
Gibt eine alternative Variable an, die für eine einzelne Zeile einer Gruppe zurückgegeben werden soll, und kann nur mit den Schlüsselwörtern MinOf, MaxOf und MedianOf eingesetzt werden.

Beschreibung

GROUPBY fasst mehrere Zeilen von Daten in einer Gruppe zusammen. Die Ausgabe dieser Funktion ist eine einzelne Spalte. Die Ausgabe ist das Ergebnis der durch Schlüsselwort angegebenen Operation bezüglich Rollup-Feld in der durch Gruppenfeld angegebenen homogenen Gruppe. Wenn es mehrere Antworten gibt, die eine bestimmte Bedingung erfüllen, wird die erste gefundene Antwort zurückgegeben.

Wenn das optionale Ausgabefeld nicht angegeben ist, ist die Ausgabe das Ergebnis der auf Rollup-Feld angewendeten Operation. Wenn Ausgabefeld angegeben ist, ist das Ergebnis das Ausgabefeld der Zeile in der Gruppe.

Wenn mehrere Zeilen in einer Gruppe die angegebene Bedingung erfüllen (etwa wenn bezüglich des Maximalwerts ein Gleichstand vorliegt), wird das Ausgabefeld zurückgegeben, das der ersten Zeile zugeordnet ist, die die Bedingung erfüllt.

Anmerkung: Wenn Sie eine mehrspaltige Gruppierung bearbeiten möchten, können Sie eine Liste von Feldnamen, die durch Kommas getrennt sind, in geschweifte Klammern "{ }" einschließen und im Aufruf des Makros GROUPBY als ersten Parameter verwenden.

Die folgenden Schlüsselwörter werden unterstützt (die Groß-/Kleinschreibung wird nicht beachtet):

Schlüsselwort	Beschreibung
Zeichenfolge?	
Ja/Nein	

CountOf	Ja	Gibt die Anzahl der Einträge in jeder Gruppe zurück (Rollup-Feld kann numerisch oder eine Zeichenfolge sein; der Rückgabewert ist unabhängig vom Wert von Rollup-Feld derselbe).
MinOf	Ja	Gibt den Minimalwert von Rollup-Feld in jeder Gruppe zurück (Rollup-Feld kann numerisch oder eine Zeichenfolge sein; wenn Rollup-Feld eine Zeichenfolge ist, wird der Wert zurückgegeben, der bei alphabetischer Sortierung dem Anfang des Alphabets am nächsten liegt).
MaxOf	Ja	Gibt den Maximalwert von Rollup-Feld in jeder Gruppe zurück (Rollup-Feld kann numerisch oder eine Zeichenfolge sein; wenn Rollup-Feld eine Zeichenfolge ist, wird der Wert zurückgegeben, der bei alphabetischer Sortierung dem Ende des Alphabets am nächsten liegt).
DiffOf	Ja	Gibt die Anzahl unterschiedlicher Werte von Rollup-Feld in jeder Gruppe zurück (Rollup-Feld kann numerisch oder eine Zeichenfolge sein).
AvgOf	Nein	Gibt den Durchschnittswert von Rollup-Feld in jeder Gruppe zurück (Rollup-Feld muss numerisch sein).
ModeOf	Ja	Gibt den Modalwert (d. h. den am häufigsten vorkommenden Wert) von Rollup-Feld in jeder Gruppe zurück (Rollup-Feld kann numerisch oder eine Zeichenfolge sein).
MedianOf	Ja	Gibt den gemittelten Wert (d. h. den Mittelwert bei Sortierung nach Rollup-Feld) von Rollup-Feld in jeder Gruppe zurück (Rollup-Feld kann numerisch oder eine Zeichenfolge sein; wenn Rollup-Feld eine Zeichenfolge ist, werden die Werte alphabetisch sortiert).

OrderOf	Ja	Gibt die Reihenfolge von Rollup-Feld in jeder Gruppe zurück (Rollup-Feld muss numerisch sein). Wenn mehrere Einträge denselben Wert haben, erhalten alle denselben Wert.
SumOf	Nein	Gibt die Summe von Rollup-Feld in jeder Gruppe zurück (Rollup-Feld muss numerisch sein).
StdevOf	Nein	Gibt die Standardabweichung von Rollup-Feld in jeder Gruppe zurück (Rollup-Feld muss numerisch sein).
IndexOf	Ja	Gibt den 1-basierten Index (nach Rollup-Feld sortiert) jedes Datensatzes zurück (Rollup-Feld kann numerisch oder eine Zeichenfolge sein). Die Sortierreihenfolge ist aufsteigend. Hinweis: Bei numerischen Feldern kann die Sortierreihenfolge von RankOf und IndexOf absteigend gemacht werden, indem dem Sortierfeld ein Minuszeichen (-) vorangestellt wird.
RankOf	Ja	Gibt die 1-basierte Kategorie (nach Rollup-Feld sortiert) zurück, in der jeder Datensatz liegt (Rollup-Feld kann numerisch oder eine Zeichenfolge sein). Die Sortierreihenfolge ist aufsteigend. Hinweis: Bei numerischen Feldern kann die Sortierreihenfolge von RankOf und IndexOf absteigend gemacht werden, indem dem Sortierfeld ein Minuszeichen (-) vorangestellt wird.

Beispiele

GROUPBY (Haushalts-ID, SumOf, Saldo)

Berechnet die Summe aller Kontostände für den jeweiligen Haushalt.

GROUPBY (Kunden-ID, MinOf, Date(Kontoeröffnungsdatum), Kontonummer)

Gibt die Kontonummer des ersten Kontos zurück, das vom Kunden eröffnet wurde.

GROUPBY_WHERE

Syntax

GROUPBY_WHERE(Gruppenfeld, Schlüsselwort, Rollup-Feld, Bedingungswert[, Ausgabefeld])

Parameter

- Gruppenfeld
Gibt die Variable an, nach der die Einträge gruppiert werden (d. h., alle identischen Werte der angegebenen Variable werden derselben Gruppe zugeordnet).
- Schlüsselwort
Gibt die zusammenfassende Rollup-Funktion an, die auf das Rollup-Feld angewendet werden soll.
- Rollup-Feld
Gibt die Variable an, die zusammengefasst werden soll.
- Bedingungswert
Ein Ausdruck, dessen Auswertung den Wert eins oder null ergibt und der angibt, welche Zeilen in die Rollup-Operation einbezogen werden sollen.
- Ausgabefeld
Gibt eine alternative Variable an, die für eine einzelne Zeile einer Gruppe zurückgegeben werden soll, und kann nur mit den Schlüsselwörtern MinOf, MaxOf und MedianOf eingesetzt werden.

Beschreibung

GROUPBY_WHERE fasst bestimmte Zeilen von Daten in einer Gruppe zusammen. Die Ausgabe dieser Funktion ist eine einzelne Spalte. Die Ausgabe ist das Ergebnis der durch Schlüsselwort angegebenen Operation bezüglich Rollup-Feld in der durch Gruppenfeld angegebenen homogenen Gruppe, gefiltert durch Bedingungswert. Es werden nur Zeilen mit dem Bedingungswert eins in die Berechnung einbezogen.

Wenn das optionale Ausgabefeld nicht angegeben ist, ist die Ausgabe das Ergebnis der auf Rollup-Feld angewendeten Operation. Wenn Ausgabefeld angegeben ist, ist das Ergebnis das Ausgabefeld der Zeile in der Gruppe.

Anmerkung: Unter „GROUPBY“ auf Seite 114 finden Sie weitere Informationen zu gültigen Werten für Schlüsselwort.

Beispiele

GROUPBY_WHERE (Haushalts-ID, SumOf, Saldo, Saldo>0)

Berechnet die Summe aller Konten mit positivem Saldo für den jeweiligen Haushalt.

GROUPBY_WHERE (Kunden-ID, AvgOf, Kaufbetrag, Date(aktuelles_Datum) - Date(Kaufdatum)<90)

Berechnet den durchschnittlichen Kaufbetrag des jeweiligen Kunden bei Einkäufen in den vergangenen 90 Tagen.

Syntax

Daten1 GT Daten2; Daten1 > Daten2

Parameter

Daten1

Der numerische Zellenbereich, der verglichen werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Die Zahlen, mit denen alle Werte in der angegebenen Spalte verglichen werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Daten2 muss mit der Anzahl der Spalten in Daten1 übereinstimmen, es sei denn, bei Daten2 handelt es sich um eine Konstante. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

GT vergleicht die beiden angegebenen Datenbereiche und gibt eine Eins zurück, wenn die Werte im ersten Dataset größer als die Werte im zweiten Dataset sind, andernfalls eine Null. GT gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten1 mit der entsprechenden Spalte von Daten2 verglichen wird (d. h., die erste Spalte von Daten1 wird mit der ersten Spalte von Daten2 verglichen, die zweite Spalte mit der zweiten Spalte usw.).

Wenn es sich bei Daten2 um eine Konstante handelt, wird jeder Wert in Daten1 mit dem betreffenden Wert verglichen. Wenn es sich bei Daten2 um eine Spalte handelt, werden die Berechnungen zeilenweise ausgeführt. Die erste Zeile von Daten1 wird mit der ersten Zeile von Daten2 verglichen, die zweite Zeile mit der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Der Operator GT kann durch ein Größer-als-Zeichen abgekürzt werden (>).

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

```
TEMP = 3 GT 4 oder TEMP = 3 > 4
```

Erstellt eine neue Spalte TEMP, die den Wert null enthält (drei ist nicht größer als vier).

TEMP = W1 > 8
Erstellt eine neue Spalte TEMP, in der jeder Wert eins ist, wenn der entsprechende Zeilenwert von Spalte W1 größer als die Zahl acht ist; andernfalls ist der Wert null.
TEMP = W1:W3 > 2
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind der mit zwei verglichene Inhalt von Spalte W1, die Werte von Spalte WX sind der mit zwei verglichene Inhalt von Spalte W2 und die Werte von Spalte WY sind der mit zwei verglichene Inhalt von Spalte W3.
TEMP = W1 > W1
Erstellt eine neue Spalte TEMP, die nur Nullen enthält (keine Zahl ist größer als sie selbst).
TEMP = W1 > W2
Erstellt eine neue Spalte TEMP, wobei jeder Zeilenwert von Spalte W1 mit dem entsprechenden Zeilenwert von Spalte W2 verglichen wird.
TEMP = W1:W3 > W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Werte von Spalte W1 mit den entsprechenden Zeilenwerten von Spalte W4 verglichen. Die Spalte WX enthält den Vergleich von Spalte W2 mit Spalte W5. Die Spalte WY enthält den Vergleich von Spalte W3 mit Spalte W6.
TEMP = W1[10:20] > W2 oder TEMP = W1[10:20] > W2[1:11]
Erstellt eine neue Spalte TEMP, deren erste 11 Zellen das Ergebnis des Vergleichs der Werte in Zeile 10-20 von Spalte W1 mit den Werten in Zeile 1-11 von Spalte W2 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
EQ	Gibt TRUE zurück, wenn der eine Datenbereich dem anderen gleich ist.
GE	Gibt TRUE zurück, wenn der eine Datenbereich größer-gleich dem anderen ist.
LE	Gibt TRUE zurück, wenn der eine Datenbereich kleiner-gleich dem anderen ist.
LT	Gibt TRUE zurück, wenn der eine Datenbereich kleiner als der andere ist.
NE	Gibt TRUE zurück, wenn der eine Datenbereich dem anderen ungleich ist.

HISTOGRAM



Syntax

HISTOGRAM(Daten, Bin-Spalte)

Parameter

Daten

Der Zellenbereich, dessen Histogramm berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Alle Spalten in Daten müssen demselben Datentyp (numerisch oder Zeichenfolge) angehören. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Bin-Spalte

Die Werte für die Bin-Grenzen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen einspaltigen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Der Datentyp von Bin-Spalte muss mit dem von Daten identisch sein. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

HISTOGRAM berechnet das Histogramm (d. h. die Häufigkeit des Vorkommens von Datenwerten in verschiedenen Bins) der Werte im angegebenen Datenbereich. HISTOGRAM gibt eine einzelne Spalte mit der Anzahl der Datenwerte in Daten zurück, die innerhalb des entsprechenden Bin-Bereichs liegen, der durch Bin-Spalte angegeben ist.

Bei numerischen Werten bilden zwei benachbarte Zeilen von Bin-Spalte einen "Bin" (d. h. einen Wertebereich). Jeder Wert in Daten, der innerhalb eines Bins liegt, wird für diesen Bin kumuliert. Die Ausgabespalte enthält die resultierende Anzahl der Datenwerte innerhalb jedes Bins. Der erste Grenzwert ist im Bin *eingeschlossen*; der zweite Grenzwert ist *ausgeschlossen*. So enthält beispielsweise das Paar der Grenzwerte 1 und 2 die Anzahl aller Werte in Daten, die größer-gleich 1 und kleiner als 2 sind. Die Länge der Ausgabespalte ist um eins kleiner als die Länge von Bin-Spalte.

Bei Zeichenfolgen werden nur exakte Übereinstimmungen der Zeichenfolge in Bin-Spalte in dem betreffenden Bin gezählt. Die Länge der Ausgabespalte ist die Länge von Bin-Spalte. Wenn Bin-Spalte bei numerischen Daten skalar ist (d. h., einen einzelnen Zellenwert enthält), wird die Anzahl der Einträge in Daten gezählt.

Anmerkung: Die Makrofunktion HISTOGRAM ordnet Dateneinträge auf andere Weise in Bins ein als der Histogrammgraph von **IBM PredictiveInsight**. Der Histogrammgraph schließt das Minimum (abgesehen vom ganz linken Bin) aus und schließt das Maximum jeder Bin-Grenze ein.

Beispiele

```
TEMP = HISTOGRAM(1...10, COLUMN(1, 3, 10))
```

Erstellt eine neue Spalte TEMP, die die Werte 2 und 7 enthält.

```
TEMP = HISTOGRAM("a", "b", "a", COLUMN("a", "b", "c"))
```

Erstellt eine neue Spalte TEMP, die die Werte 2, 1 und 0 enthält.

TEMP = HISTOGRAM(W1, COLUMN(1, 25, 50, 75, 101))
Erstellt eine neue Spalte TEMP, die vier Werte enthält. Der erste Wert ist die Anzahl der Werte in Spalte W1, die größer-gleich 1 und kleiner als 25 sind. Der zweite Wert ist die Anzahl der Werte in Spalte W1, die größer-gleich 25 und kleiner als 50 sind. Der dritte und der vierte Wert enthalten die Anzahl im dritten bzw. im vierten Quartil.
TEMP = HISTOGRAM(W1:W3, W4)
Erstellt eine neue Spalte TEMP, wobei jeder Wert die Anzahl der Werte in Spalte W1 bis W3 darstellt, die innerhalb der durch Spalte W4 angegebenen Bin-Grenzen liegen.
TEMP = HISTOGRAM(W1[50:100]:W5, W6[1:10])
Erstellt eine neue Spalte TEMP, die 10 Werte enthält. Jeder Wert ist die Anzahl der Datenwerte in Zeile 50-100 von Spalte W1 bis W5, die innerhalb der durch Zeile 1-10 von Spalte W6 angegebenen Bin-Grenzen liegen.

Zugehörige Funktionen

Funktion	Beschreibung
COUNT	Zählt die Anzahl der Zellen, die Werte enthalten, im angegebenen Datenbereich.

IF

Syntax

IF(Prädikatspalte, then-Wert); IF(Prädikatspalte, then-Wert, else-Wert)

Parameter

Prädikatspalte

Eine Spalte mit booleschen Werten oder ein Ausdruck, dessen Auswertung eine einzelne Spalte mit booleschen Werten ergibt. Boolesche Werte werden als null oder ungleich null interpretiert. Diese Spalte sollte mindestens so viele Zeilen enthalten wie der Datenbereich, aus dem die Daten extrahiert werden. Andernfalls wird Prädikatspalte auf die Anzahl der Zeilen begrenzt, die von der Makrofunktion EXTRACT verarbeitet werden (siehe die folgende "Beschreibung").

then-Wert

Der oder die Werte, die zurückgegeben werden, wenn die entsprechende Zeile von Prädikatspalte einen Wert ungleich null enthält. Dabei kann es sich um einen konstanten Wert, eine Spalte oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Unter „Makrofunktionsparameter“ auf Seite 9 finden Sie weitere Informationen zur Formatdefinition von then-Wert (identisch mit Daten).

else-Wert

Wenn dieser optionale Parameter angegeben wird, wird dieser Wert zurückgegeben, falls die entsprechende Zeile von Prädikatspalte eine Null enthält. Dabei kann es sich um einen konstanten Wert, eine Spalte oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Wenn else-Wert nicht angegeben wird, wird jedes Mal eine Null zurückgegeben, wenn die Auswertung von Prädikatspalte 'falsch' ergibt. Unter „Makrofunktionsparameter“ auf Seite 9 finden

Sie weitere Informationen zur Formatdefinition von else-Wert (identisch mit Daten).

Beschreibung

IF wertet den Ausdruck in Prädikatspalte aus und gibt then-Wert zurück, wenn der Ausdruck wahr ist, bzw. else-Wert, wenn der Ausdruck falsch ist. In then-Wert und in else-Wert wird dieselbe Anzahl von Spalten zurückgegeben. Die neuen Spalten enthalten die entsprechenden Zeilenwerte aus then-Wert, wenn der Wert von Prädikatspalte ungleich null ist. Wenn else-Wert angegeben wird, wird dieser Wert zurückgegeben, falls der Wert von Prädikatspalte null ist. Wenn else-Wert nicht angegeben wird, wird null zurückgegeben.

Da IF zeilenweise vorgeht, wird für jede Zeile bis zum letzten Wert der kürzesten Spalte (d. h. der kürzesten Spalte von Prädikatspalte, then-Wert und else-Wert) ein Ergebnis erzeugt.

Anmerkung: In der Regel empfiehlt es sich, eine Prädikatspalte zu erstellen, die eine der Vergleichsmakrofunktionen verwendet (==, >, <, ISEVEN, ISODD usw.).

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = IF(1, W1)	Erstellt eine neue Spalte TEMP, die eine Kopie von Spalte W1 enthält.
TEMP = IF(W1, 1, 0)	Erstellt eine neue Spalte TEMP, in der jeder Wert eins ist, wenn der entsprechende Zeilenwert von Spalte W1 ungleich null ist; andernfalls ist der Wert null.
TEMP = IF(W3, W1, W2)	Erstellt eine neue Spalte TEMP, in der jeder Wert aus Spalte W1 kopiert wird, wenn der entsprechende Wert von Spalte W3 ungleich null ist; andernfalls wird der Wert aus Spalte W2 kopiert.
TEMP = IF(ABS(W1-AVG(W1)) < STDV(W1), W1)	Erstellt eine neue Spalte TEMP, die jeden Wert in Spalte W1 enthält, der weniger als eine Standardabweichung vom Mittelwert entfernt ist.
TEMP = IF(W3[20:30], W1[30:40], W2)	Erstellt eine neue Spalte TEMP, die Werte für Zeile 10-20 enthält. Jeder Wert wird aus Spalte W1 (Zeile 10-20) kopiert, wenn der entsprechende Wert von Spalte W3 (Zeile 30-40) ungleich null ist; andernfalls wird der Wert aus Spalte W2 (Zeile 1-11) kopiert.

Zugehörige Funktionen

Funktion	Beschreibung
EXTRACT	Extrahiert Zeilen aufgrund der Werte in einer Prädikatspalte.
SELECT	Gibt die angegebene(n) Spalte(n) aus einem Datenbereich zurück.

IN

Syntax

Werte IN (Wert1 AND Wert2) oder Werte IN Unterabfrage

Parameter

Die erste Form ermöglicht die Verwendung einer Werteliste statt einer Unterabfrage.

Die zweite Form verwendet eine Unterabfrage, deren Auswertung ein Zwischenergebnis erzeugt, auf das weitere Verarbeitungsschritte angewendet werden können.

Beschreibung

Das Prädikat IN ermöglicht die Verwendung einer Liste von Werten statt einer Unterabfrage oder es leitet eine Unterabfrage ein.

Anmerkung: IN unterscheidet sich von ISMEMBER: IN wird nach Möglichkeit in der Datenbank ausgeführt, während ISMEMBER auf dem Server berechnet wird.

Anmerkung: Das Prädikat IN verfügt über eine negative Version, NOT IN. Deren Format ist mit dem von IN identisch. NOT IN ist nur wahr, wenn der angegebene Wert nicht in den von der Unterabfrage zurückgegebenen Werten gefunden wird.

Dieses Makro ist in IBM Interact verfügbar.

Wichtig: Wenn Sie IN in IBM Interact einsetzen, können Sie nur die Syntax Wert IN (Wert1 AND Wert2) verwenden.

Beispiele

```
TEMP = IN(25, COLUMN(1...10))
```

Gibt die angegebene(n) Spalte(n) aus einem Datenbereich zurück.

```
TEMP = IN("Katze", COLUMN("Katze", "Hund", "Vogel"))
```

Erstellt eine neue Spalte TEMP, die den Wert eins enthält.

```
TEMP = IN(W1, W1)
```

Erstellt eine neue Spalte TEMP, die nur Einsen enthält.

```
TEMP = IN(W1, W2)
```

Erstellt eine neue Spalte TEMP, in der jeder Wert eins ist, wenn die entsprechende Zeile von Spalte W1 einen Wert in Spalte W2 enthält; andernfalls ist der Wert null.

Beispiele

```
SELECT Titel, aktueller_Leihpreis FROM Filmtitel WHERE unsere_Kosten IN (14.95, 24.95, 29.95) ;
```

Dies bedeutet, dass alle Filme, die 14,95 €, 24,95 € oder 29,95 € kosten, für das Prädikat IN als 'wahr' bewertet werden; somit wird eine Liste der Titel und der zugehörigen aktuellen Leihpreise für die betreffenden Filme erstellt.

```
UPDATE Filmtitel SET aktueller_Leihpreis = (regulärer_Leihpreis * .9) WHERE Titel
IN ( SELECT Filmtitel FROM Filmstars WHERE Schauspieler_Nachname = 'Stewart' AND
Schauspieler_Vorname = 'James' ) ;
```

Nachdem die Unterabfrage ihre Liste der Filmtitel erstellt hat, wird die Liste anhand der Tabelle FILMTITEL verarbeitet und die jeweiligen Leihpreise für *Die Nacht vor der Hochzeit*, *Ist das Leben nicht schön?* usw. werden ermäßigt.

INIT



Syntax

```
INIT(Wert1 [, Wert2]...); INIT(Spalte)
```

Parameter

Wert1

Der numerische Wert der rekursiven Funktion zur Zeit ($t-1$).

Wert2

Der numerische Wert der rekursiven Funktion zur Zeit ($t-n$); dabei ist n die Parameterzahl. Dieser Parameter kann mehrmals wiederholt werden, um Anfangswerte für eine unbegrenzte Anzahl vorheriger Zeitschritte festzulegen.

Spalte

Eine Spalte mit numerischen Werten. Die erste Zelle wird dem Zeitschritt ($t-1$) zugeordnet, die zweite Zelle dem Zeitschritt ($t-2$) usw.

Beschreibung

INIT gibt die Anfangswerte für die Definition einer rekursiven Funktion an. Der erste angegebene Wert wird dem Zeitschritt ($t-1$) zugeordnet, der zweite Wert dem Zeitschritt ($t-2$) usw. Wenn ein Zeitschritt nicht mit INIT initialisiert wird, wird sein Wert als null vorausgesetzt. So ist zum Beispiel in der Anweisung

```
W1 = INIT(1, 2, 3)
```

der Wert für Zeitschritt ($t-4$) null (ebenso wie für alle übrigen Zeitschritte, die weiter in der Vergangenheit liegen). Vor der Definition einer rekursiven Funktion ist eine INIT-Anweisung erforderlich.

Anmerkung: Um alle Werte als null zu initialisieren, können Sie einfach INIT() ohne Argumente angeben.

Die Makrofunktion INIT gibt keine Werte zurück. Wenn sie allein in einer Funktionsdefinition eingesetzt wird, gibt sie eine leere Spalte zurück.

Beispiele

```
TEMP = INIT() t = 1 TO 10 TEMP = 1 + TEMP[t-1]
```

Erstellt eine neue Spalte TEMP, die die Werte 1-10 enthält.

```
TEMP = INIT(1) t = 1 TO 100 TEMP = TEMP[t-1]+TEMP[t-1]
```

Erstellt eine neue Spalte TEMP, die die Werte 2, 4, 8, 16, 32 usw. enthält. Die ersten 100 Zellen von TEMP enthalten Werte.

```
TEMP = INIT(1, 2, 3, 4, 5) t = 1 TO 500 TEMP = TEMP[t-5]
```

Erstellt eine Spalte TEMP, die die Wertefolge 5, 4, 3, 2, 1 in 100-facher Wiederholung enthält.

```
TEMP = INIT(1, 2, 3) t = 1 TO 1000 TEMP = 2*TEMP[t-1] + 4*TEMP[t-2]^2 - TEMP[t-3]
```

Erstellt eine neue Spalte TEMP, die Werte für die kursive Funktion enthält:

$$\text{TEMP} = 2 * \text{TEMP}(t - 1) + 4 * \text{TEMP}(t - 2)^2 - \text{TEMP}(t - 2)$$

Es werden 1000 Zellenwerte berechnet.

Zugehörige Funktionen

Funktion	Beschreibung
T0	Operator zur Bereichsgenerierung.

INT

Syntax

INT(Daten)

Parameter

Daten

Die numerischen Werte, die auf einen Ganzzahlwert abgerundet werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

INT berechnet die größte Ganzzahl, die kleiner als die Werte im angegebenen Datenbereich ist (gleichbedeutend mit der Untergrenze). INT gibt für jede Eingabespalte eine neue Spalte zurück, die die Untergrenze der Zahlen in der entsprechenden Eingabespalte enthält.

Anmerkung: INT ist mit der Makrofunktion FLOOR identisch.

Beispiele

TEMP = INT(4,7)
Erstellt eine neue Spalte TEMP, die den Wert 4 enthält.
TEMP = INT(-1,5)
Erstellt eine neue Spalte TEMP, die den Wert -2 enthält.
TEMP = INT(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert die größte Ganzzahl kleiner-gleich dem Inhalt von Spalte W1 darstellt.
TEMP = W1 - INT(W1)
Erstellt eine neue Spalte TEMP, die den Dezimalanteil jedes Werts in Spalte W1 enthält.
TEMP = INT(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils die größten Ganzzahlen kleiner-gleich dem Inhalt von Spalte W1, die Werte von Spalte WX sind die größten Ganzzahlen kleiner-gleich dem Inhalt von Spalte W2 und die Werte von Spalte WY sind die größten Ganzzahlen kleiner-gleich dem Inhalt von Spalte W3.
TEMP = INT(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen die größten Ganzzahlen kleiner-gleich den entsprechenden Werten in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = INT(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind die größten Ganzzahlen kleiner-gleich den entsprechenden Zeilenwerten von Spalte W1, die Werte in der Spalte WX sind die größten Ganzzahlen kleiner-gleich den entsprechenden Zeilenwerten von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
ROUND	Berechnet den gerundeten Wert des Inhalts des angegebenen Datenbereichs.
TRUNCATE	Berechnet den Ganzzahlanteil jedes Werts im angegebenen Datenbereich.

INTEGRAL



Syntax

INTEGRAL(Daten [, Multiplikator])

Parameter

Daten

Die numerischen Werte, deren Integral berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck

handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Multiplikator

Ein Wert, mit dem jeder Wert in Daten multipliziert werden soll. Dabei kann es sich um einen konstanten Wert oder um einen Ausdruck handeln, dessen Auswertung eine Konstante ergibt.

Beschreibung

INTEGRAL berechnet das Integral der Werte in einer Zeitreihe. Jeder Wert ist die Summe aller vorherigen Werte in der Zeit. Wenn für Multiplikator ein Wert angegeben ist, wird jeder Wert mit dem angegebenen Wert multipliziert. INTEGRAL gibt für jede Eingabespalte eine neue Spalte zurück, die das Integral der Werte in der entsprechenden Eingabespalte enthält.

Beispiele

TEMP = INTEGRAL(5)
Erstellt eine neue Spalte TEMP, die den Wert 5 enthält.
TEMP = INTEGRAL(COLUMN(1,2,3))
Erstellt eine neue Spalte TEMP, die die Werte 1, 3 und 6 enthält.
TEMP = INTEGRAL(COLUMN(1,2,3), 2)
Erstellt eine neue Spalte TEMP, die die Werte 2, 6 und 12 enthält.
TEMP = INTEGRAL(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert die Summe aller vorherigen Zellen in Spalte W1 darstellt.
TEMP = INTEGRAL(W1, 10)
Erstellt eine neue Spalte TEMP, in der jeder Wert die Summe aller vorherigen Zellen in Spalte W1 mal 10 darstellt.
TEMP = INTEGRAL(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die Integrale von Spalte W1, die Werte von Spalte WX sind die Integrale von Spalte W2 und die Werte von Spalte WY sind die Integrale von Spalte W3.
TEMP = INTEGRAL(W1:W3)
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils das Integral der entsprechenden Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = INTEGRAL(W1:W3)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind die Integrale der entsprechenden Zeilenwerte von Spalte W1, die Werte in der Spalte WX sind die Integrale der entsprechenden Zeilenwerte von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
DERIVATIVE	Berechnet das abgeleitete Produkt der Werte im angegebenen Datenbereich.
SUM oder TOTAL	Berechnet die Summe eines Zellenbereichs.

INVERSE

Syntax

INVERSE(Daten)

Parameter

Daten

Die numerischen Werte, deren Umkehrfunktion berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

INVERSE berechnet den Negativwert der Werte im angegebenen Datenbereich. INVERSE gibt $-x$ zurück (d. h., negative Werte werden als positive Werte zurückgegeben, positive Werte werden als negative Werte zurückgegeben). INVERSE gibt für jede Eingabespalte eine neue Spalte zurück, die die Umkehrfunktion der Werte in der entsprechenden Eingabespalte enthält.

Anmerkung: Stellen Sie zum Umkehren eines Werts oder einer Spalte ein Minuszeichen (-) voran. Beispielsweise ist $W2 = -W1$ identisch mit $W2 = \text{INVERSE}(W1)$.

Beispiele

TEMP = INVERSE(3,2)
Erstellt eine neue Spalte TEMP, die den Wert -3,2 enthält.
TEMP = INVERSE(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert den Negativwert der Werte in Spalte W1 darstellt.
TEMP = INVERSE(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der Negativwert der Werte in Spalte W1, die Werte von Spalte WX sind der Negativwert der Werte in Spalte W2 und die Werte von Spalte WY sind der Negativwert der Werte in Spalte W3.
TEMP = INVERSE(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen die negativen Werte der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.

```
TEMP = INVERSE(W1[1:5]:W2)
```

Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind die negativen Werte der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind die negativen Werte der entsprechenden Zeilen von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
ABS	Berechnet den absoluten Wert des Inhalts des angegebenen Datenbereichs.
NOT	Berechnet das logische Nicht des Inhalts des angegebenen Datenbereichs.
SIGN	Berechnet das Vorzeichen (positiv oder negativ) der Werte im angegebenen Datenbereich.

IS

Syntax

IS <Schlüsselwort>

Parameter

Schlüsselwort

Suchbedingung, üblicherweise "NULL," "TRUE," "UNKNOWN" oder "FALSE".

Beschreibung

IS wird in komplexen Suchbedingungen eingesetzt. Je komplexer die Suche, desto nützlicher kann die IS-Bedingung sein. Diese booleschen Suchbedingungen stellen eine alternative Möglichkeit dar, einfache Suchbedingungen zu formulieren.

Dieses Makro ist in IBM Interact verfügbar.

IS gibt in IBM Interact andere Ergebnisse zurück als in IBM Kampagne. Bei NULL wird 1 zurückgegeben, wenn mindestens ein Nullwert für eine Zielgruppen-ID vorliegt. Bei UNKNOWN wird für eine Zielgruppen-ID 1 zurückgegeben, wenn sie keinen Wert aufweist.

Beispiele

```
SELECT Kunde FROM Kundentabelle1 WHERE (Nachname = "Schmidt" AND Vorname = "Hans")  
IS TRUE ;
```

erzeugt eine Liste aller Kunden mit dem Namen Hans Schmidt.

```
SELECT Kunde FROM Kundentabelle1 WHERE (Nachname = "X" AND Vorname = "X") IS  
UNKNOWN ;
```

sucht nach Werten ungleich null.

```
SELECT Kosten FROM Kostentabelle1 WHERE (aktuelle_Kosten = "200" IS FALSE ;  
listet alle Werte aus der Kostentabelle auf, die von 200 € verschieden sind.
```

ISERROR

Syntax

ISERROR(Daten)

Parameter

Daten

Die Werte, für die geprüft werden soll, ob in den Zeilen ein Fehler (d. h. eine ???-Zelle) enthalten ist. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

ISERROR prüft, ob eine der Zellen in jeder Zeile des angegebenen Datenbereichs einen Fehler (d. h. eine ???-Zelle) enthält. ISERROR gibt eine neue Spalte zurück, in der jede Zeile eine Eins enthält, wenn die entsprechende Zeilen von Daten einen Fehler enthält. Andernfalls enthält die Zeile eine Null. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der längsten Spalte.

Anmerkung: Mit dieser Funktion können Fehler in einer Spalte aufgespürt werden, um anschließend mit der Makrofunktion EXTRACT die korrekten Datenzeilen zu extrahieren.

Beispiele

```
TEMP = ISERROR(-3)
```

Erstellt eine neue Spalte TEMP, die den Wert null enthält.

```
TEMP = ISERROR(W1)
```

Erstellt eine neue Spalte TEMP, in der jeder Wert eins ist, wenn die entsprechende Zeile von Spalte W1 die Angabe ??? enthält; andernfalls ist der Wert null.

```
TEMP = ISERROR(W1:W3)
```

Erstellt eine neue Spalte TEMP, in der jeder Wert eins ist, wenn eine der Zellen in den entsprechenden Zeilen von Spalte W1 bis W3 die Angabe ??? enthält; andernfalls ist der Wert null.

```
TEMP = ISERROR(W1[50:100]:W10)
```

Erstellt eine neue Spalte TEMP mit Werten in Zeile 1-50. Jeder Wert ist eins, wenn eine der Zellen in Zeile 50-100 von Spalte W1 bis W10 die Angabe ??? enthält; andernfalls ist der Wert null.

Zugehörige Funktionen

Funktion	Beschreibung
EXTRACT	Extrahiert Zeilen aufgrund der Werte in einer Prädikatspalte.

ISEVEN

Syntax

ISEVEN(Daten)

Parameter

Daten

Die numerischen Werte, bei denen geprüft werden soll, ob sie gerade sind. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

ISEVEN testet jeden Wert im angegebenen Dataset auf Geradheit. ISEVEN gibt für jede Eingabespalte eine neue Spalte zurück, die bei allen geraden Werten (d. h., der Wert modulo zwei ist null) eine Eins und bei allen ungeraden Werten eine Null enthält.

Anmerkung: Bei Werten, die keine Ganzzahlen sind, wird zunächst die Makrofunktion INT angewendet. Beispiel: ISEVEN(2,5) = 1, da 2 gerade ist.

Beispiele

TEMP = ISEVEN(-3)
Erstellt eine neue Spalte TEMP, die den Wert null enthält.
TEMP = ISEVEN(MERGE(3, 2, 0))
Erstellt drei neue Spalten TEMP, WX und WY. TEMP enthält den Wert 0, WX enthält den Wert 1 und WY enthält den Wert 1.
TEMP = ISEVEN(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert das Ergebnis der Prüfung des Inhalts von Spalte W1 auf Geradheit darstellt.
TEMP = ISEVEN(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind das Ergebnis der Prüfung des Inhalts von Spalte W1 auf Geradheit, die Werte von Spalte WX sind das Ergebnis der Prüfung des Inhalts von Spalte W2 auf Geradheit und die Werte von Spalte WY sind das Ergebnis der Prüfung des Inhalts von Spalte W3 auf Geradheit.
TEMP = ISEVEN(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen die Ergebnisse der Prüfung der Werte in Zeile 10-20 von Spalte W1 auf Geradheit enthalten. Die anderen Zellen in TEMP sind leer.

```
TEMP = ISEVEN(W1[1:5]:W2)
```

Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind die Ergebnisse der Prüfung der entsprechenden Zeilen von Spalte W1 auf Geradheit, die Werte in der Spalte WX sind die Ergebnisse der Prüfung der entsprechenden Zeilen von Spalte W2 auf Geradheit.

Zugehörige Funktionen

Funktion	Beschreibung
ISODD	Prüft, ob die Eingabewerte ungerade (d. h. nicht durch zwei teilbar) sind.
ISMEMBER	Prüft einen Eingabebereich anhand einer "Tabelle" von Werten und gibt eine Eins zurück, wenn ein Wert in der Tabelle enthalten ist, andernfalls eine Null.

ISMEMBER



Syntax

ISMEMBER(Daten, Tabelle)

Parameter

Daten

Die numerischen Werte, bei denen geprüft werden soll, ob sie Elemente einer Tabelle sind. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Wenn Daten mehrere Spalten enthält, müssen alle Spalten demselben Datentyp angehören (entweder numerische Werte oder Zeichenfolgen). Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Tabelle

Die Tabellenwerte für den Vergleich. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Wenn Tabelle mehrere Spalten enthält, müssen alle Spalten demselben Datentyp wie Daten angehören (entweder numerische Werte oder Zeichenfolgen). Die Anzahl der Datenwerte in Tabelle darf 16 Millionen nicht überschreiten. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Anmerkung: ISMEMBER unterscheidet sich von IN: ISMEMBER wird auf dem Server berechnet, während IN nach Möglichkeit in der Datenbank berechnet wird.

Beschreibung

ISMEMBER vergleicht die Datenwerte im angegebenen Datenbereich mit einer Tabelle von Datenwerten. ISMEMBER gibt für jede Eingabespalte in Daten eine neue Spalte zurück, die jeweils eine Eins enthält, wenn der entsprechende Eingabewert ein Element von Tabelle ist; andernfalls enthält die Spalte eine Null.

Beispiele

TEMP = ISMEMBER(25, COLUMN(1...10))
Erstellt eine neue Spalte TEMP, die den Wert null enthält.
TEMP = ISMEMBER("Katze", COLUMN("Katze", "Hund", "Vogel"))
Erstellt eine neue Spalte TEMP, die den Wert eins enthält.
TEMP = ISMEMBER(W1, W1)
Erstellt eine neue Spalte TEMP, die nur Einsen enthält.
TEMP = ISMEMBER(W1, W2)
Erstellt eine neue Spalte TEMP, in der jeder Wert eins ist, wenn die entsprechende Zeile von Spalte W1 einen Wert in Spalte W2 enthält; andernfalls ist der Wert null.
TEMP = ISMEMBER(W1:W2, W5:W10)
Erstellt zwei neue Spalten TEMP und WX. Die Spalte TEMP enthält eine Eins, wenn die entsprechende Zeile von Spalte W1 ein Element der Spalten W5 bis W10 ist, andernfalls eine Null. Die Spalte WX enthält eine Eins, wenn die entsprechende Zeile von Spalte W2 ein Element der Spalten W5 bis W10 ist, andernfalls eine Null.
TEMP = ISMEMBER(W1[10:15]:W2, W3[1:100]:W6)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-6 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind eins, wenn die Inhalte der Zeilen 10-15 von Spalte W1 Elemente der Zeilen 1-100 von Spalte W3 bis W6 sind. Die Werte in der Spalte WX sind eins, wenn die Inhalte der Zeilen 10-15 von Spalte W2 Elemente der Zeilen 1-100 von Spalte W3 bis W6 sind.

Zugehörige Funktionen

Funktion	Beschreibung
ISEVEN	Prüft, ob die Eingabewerte gerade (d. h. durch zwei teilbar) sind.
ISODD	Prüft, ob die Eingabewerte ungerade (d. h. nicht durch zwei teilbar) sind.

ISODD

Syntax

ISODD(Daten)

Parameter

Daten

Die numerischen Werte, bei denen geprüft werden soll, ob sie ungerade sind. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

ISODD testet jeden Wert im angegebenen Dataset auf Ungeradheit. ISODD gibt für jede Eingabespalte eine neue Spalte zurück, die bei allen ungeraden Werten (d. h., der Wert modulo zwei ist eins) eine Eins und bei allen geraden Werten eine Null enthält.

Anmerkung: Bei Werten, die keine Ganzzahlen sind, wird zunächst die Makrofunktion INT angewendet. Beispiel: ISODD(2,5) = 0, da 2 nicht ungerade ist.

Beispiele

TEMP = ISODD(-3)
Erstellt eine neue Spalte TEMP, die den Wert eins enthält.
TEMP = ISODD(MERGE(1, 4, 0))
Erstellt drei neue Spalten TEMP, WX und WY. TEMP enthält den Wert 1, WX enthält den Wert 0 und WY enthält den Wert 0.
TEMP = ISODD(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert das Ergebnis der Prüfung des Inhalts von Spalte W1 auf Ungeradheit darstellt.
TEMP = ISODD(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind das Ergebnis der Prüfung des Inhalts von Spalte W1 auf Ungeradheit, die Werte von Spalte WX sind das Ergebnis der Prüfung des Inhalts von Spalte W2 auf Ungeradheit und die Werte von Spalte WY sind das Ergebnis der Prüfung des Inhalts von Spalte W3 auf Ungeradheit.
TEMP = ISODD(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen die Ergebnisse der Prüfung der Werte in Zeile 10-20 von Spalte W1 auf Ungeradheit enthalten. Die anderen Zellen in TEMP sind leer.
TEMP = ISODD(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind die Ergebnisse der Prüfung der entsprechenden Zeilen von Spalte W1 auf Ungeradheit, die Werte in der Spalte WX sind die Ergebnisse der Prüfung der entsprechenden Zeilen von Spalte W2 auf Ungeradheit.

Zugehörige Funktionen

Funktion	Beschreibung
ISEVEN	Prüft, ob die Eingabewerte gerade (d. h. durch zwei teilbar) sind.
ISMEMBER	Prüft einen Eingabebereich anhand einer "Tabelle" von Werten und gibt eine Eins zurück, wenn ein Wert in der Tabelle enthalten ist, andernfalls eine Null.

KURTOSIS



Syntax

KURTOSIS(Daten [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, deren Kurtosis berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt. In Daten müssen mindestens vier Werte enthalten sein.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabedatenbereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL

Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL

Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW

Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

KURTOSIS berechnet die Kurtosis der Werte im angegebenen Datenbereich. Kurtosis ist ein Messung für die relative Steilheit oder Flachheit einer Verteilung im Vergleich zu einer Normalverteilung. Je negativer die Kurtosis ist, desto flacher ist die Verteilung. Je positiver die Kurtosis ist, desto ausgeprägter ist der Gipfel der Verteilung.

Die Kurtosis wird wie folgt berechnet:

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

Dabei steht n für die Anzahl der Stichproben in der Verteilung, *Mittelwert* für den Durchschnitt und σ für die Standardabweichung der Verteilung. Es müssen mindestens drei Datenwerte angegeben werden, um die Kurtosis berechnen zu können.

Beispiele

TEMP = KURTOSIS(MERGE(3, 4, 5)) oder TEMP = KURTOSIS(MERGE(3, 4, 5), ALL)
Erstellt eine neue Spalte TEMP, die den Wert -1,5 enthält.
TEMP = KURTOSIS(W1)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Kurtosis des Inhalts von Spalte W1 darstellt.
TEMP = KURTOSIS(W1:W3)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Kurtosis des Inhalts der Spalten W1, W2 und W3 darstellt.
TEMP = KURTOSIS(W1[10:20])
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Kurtosis der Zellen in Zeile 10-20 von Spalte W1 darstellt.
TEMP = KURTOSIS(W1[1:5]:W4)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Kurtosis der Zellen in Zeile 1-5 von Spalte W1 bis W4 darstellt.
TEMP = KURTOSIS(W1:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY. Der Einzelwert in der Spalte TEMP ist die Kurtosis des Inhalts von Spalte W1, der Einzelwert in der Spalte WX ist die Kurtosis des Inhalts von Spalte W2 und der Einzelwert in der Spalte WY ist die Kurtosis des Inhalts von Spalte W3.
TEMP = KURTOSIS(MERGE(1,4), COL)
Erstellt zwei neue Spalten TEMP und WX, die beide den Wert -3 enthalten.
TEMP = KURTOSIS(W1[1:5]:W3, COL) oder TEMP = KURTOSIS(W1[1:5]:W3[1:5], COL)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils einen Einzelwert enthalten. Der Wert in der Spalte TEMP ist die Kurtosis der Zellen in Zeile 1-5 von Spalte W1, der Wert in der Spalte WX ist die Kurtosis der Zellen in Zeile 1-5 von Spalte W2 und der Wert in der Spalte WY ist die Kurtosis der Zellen in Zeile 1-5 von Spalte W3.
TEMP = KURTOSIS(W1:W3, ROW)
Erstellt eine neue Spalte TEMP, in der jeder Zelleneintrag die Kurtosis der entsprechenden Zeile in Spalte W1, W2 und W3 darstellt.
TEMP = KURTOSIS(W1[1:5]:W3, ROW) oder TEMP = KURTOSIS(W1[1:5]:W3[1:5], ROW)
Erstellt eine neue Spalte TEMP, in der die Zellen in Zeile 1-5 die Kurtosis der entsprechenden Zeile in Spalte W1 bis W3 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
STAT	Berechnet das erste bis vierte Moment des angegebenen Datenbereichs.
SKEW	Berechnet die Schiefe der Verteilung eines Zellenbereichs.
STDV oder STDEV	Berechnet die Standardabweichung eines Zellenbereichs.
VARIANCE	Berechnet die Varianz eines Zellenbereichs.

LAG



Syntax

LAG(Verzögerung, Daten)

Parameter

Verzögerung

Die Anzahl der für die Verzögerung vorgesehenen Zeitschritte. Dieser Wert muss eine positive Ganzzahl sein.

Daten

Die Werte, die verzögert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

LAG gibt Werte im Eingabedatenbereich zurück, die um die angegebene Anzahl von Zeitschritten verzögert sind. Jede Eingabespalte wird als Datenserie in der Zeit betrachtet. LAG gibt für jede Eingabespalte eine neue Spalte zurück. Jede neue Spalte enthält die zeitverzögerten Werte (die Anzahl der Zeitschritte wird durch Verzögerung angegeben) der Zahlen in der entsprechenden Eingabespalte. Die ersten Verzögerung Werte in den zurückgegebenen Spalten sind null. Die Länge der zurückgegebenen Spalte(n) ist die Länge der entsprechenden Eingabespalte(n) + Verzögerung.

Anmerkung: Die Makrofunktion LAG gibt eine Spalte mit Werten zurück, sodass Zelle WY[x] = Daten[x - Verzögerung] für x > Verzögerung, andernfalls 0.

Beispiele

<code>TEMP = LAG(1, COLUMN(1,2,3,4))</code>
Erstellt eine neue Spalte TEMP, die in Zeile 1-5 den Wert 0, 1, 2, 3 bzw. 4 enthält.
<code>TEMP = LAG(2, W1)</code>
Erstellt eine neue Spalte TEMP, in der jeder Wert den Inhalt von Spalte W1 darstellt, der um zwei Zeitschritte verzögert ist.
<code>TEMP = LAG(10, W1:W3)</code>
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind der um zehn Zeitschritte verzögerte Inhalt von Spalte W1, die Werte von Spalte WX sind der um zehn Zeitschritte verzögerte Inhalt von Spalte W2 und die Werte von Spalte WY sind der um zehn Zeitschritte verzögerte Inhalt von Spalte W3.
<code>TEMP = LAG(5, W1[10:20])</code>
Erstellt eine neue Spalte TEMP, in der die ersten 5 Zellen den Wert null und die folgenden Zellen die Werte aus Zeile 10-20 von Spalte W1 enthalten. Die anderen Zellen in TEMP sind leer.
<code>TEMP = LAG(2, W1[1:5]:W2)</code>
Erstellt zwei neue Spalten TEMP und WX, die jeweils Werte in Zeile 1-7 enthalten (die anderen Zellen sind leer). Die Werte in Zeile 1-2 jeder Spalte sind null. Die übrigen Werte in der Spalte TEMP sind die Werte in Zeile 1-5 von Spalte W1. Die übrigen Werte in der Spalte WX sind die Werte in Zeile 1-5 von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
DELAY	Gibt die Verzögerung der Eingabespaltenwerte um eine angegebene Anzahl von Zeitschritten an.
SLIDE_WINDOW	Erstellt ein Muster aus einem angegebenen Fenster und schiebt es nach unten, um das nächste Muster zu erstellen.

LE

Syntax

Daten1 LE Daten2; Daten1 <= Daten2

Parameter

Daten1

Der numerische Zellenbereich, der verglichen werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Die Zahlen, mit denen alle Werte in der angegebenen Spalte verglichen werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

LE vergleicht die beiden angegebenen Datenbereiche und gibt eine Eins zurück, wenn die Werte im ersten Dataset kleiner-gleich den Werten im zweiten Dataset sind, andernfalls eine Null. LE gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten1 mit der entsprechenden Spalte von Daten2 verglichen wird (d. h., die erste Spalte von Daten1 wird mit der ersten Spalte von Daten2 verglichen, die zweite Spalte mit der zweiten Spalte usw.).

Wenn es sich bei Daten2 um eine Konstante handelt, wird jeder Wert in Daten1 mit dem betreffenden Wert verglichen. Wenn es sich bei Daten2 um eine Spalte handelt, werden die Berechnungen zeilenweise ausgeführt. Die erste Zeile von Daten1 wird mit der ersten Zeile von Daten2 verglichen, die zweite Zeile mit der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Der Operator LE kann durch ein Kleiner-als-Zeichen, auf das ein Gleichheitszeichen folgt, abgekürzt werden (<=).

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = 4 LE 4 oder TEMP = 4 <= 4
Erstellt eine neue Spalte TEMP, die den Wert eins enthält (vier ist gleich sich selbst).
TEMP = W1 <= 8
Erstellt eine neue Spalte TEMP, in der jeder Wert eins ist, wenn der entsprechende Zeilenwert von Spalte W1 kleiner-gleich der Zahl acht ist; andernfalls ist der Wert null.
TEMP = W1:W3 <= 2
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind der mit zwei verglichene Inhalt von Spalte W1, die Werte von Spalte WX sind der mit zwei verglichene Inhalt von Spalte W2 und die Werte von Spalte WY sind der mit zwei verglichene Inhalt von Spalte W3.
TEMP = W1 <= W1
Erstellt eine neue Spalte TEMP, die nur Einsen enthält (jede Zahl ist gleich sich selbst).
TEMP = W1 <= W2
Erstellt eine neue Spalte TEMP, wobei jeder Zeilenwert von Spalte W1 mit dem entsprechenden Zeilenwert von Spalte W2 verglichen wird.
TEMP = W1[10:20] <= W2 oder TEMP = W1[10:20] <= W2[1:11]
Erstellt eine neue Spalte TEMP, deren erste 11 Zellen das Ergebnis des Vergleichs der Werte in Zeile 10-20 von Spalte W1 mit den Werten in Zeile 1-11 von Spalte W2 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
EQ	Gibt TRUE zurück, wenn der eine Datenbereich dem anderen gleich ist.
GE	Gibt TRUE zurück, wenn der eine Datenbereich größer-gleich dem anderen ist.
GT	Gibt TRUE zurück, wenn der eine Datenbereich größer als der andere ist.
LT	Gibt TRUE zurück, wenn der eine Datenbereich kleiner als der andere ist.
NE	Gibt TRUE zurück, wenn der eine Datenbereich dem anderen ungleich ist.

LIKE

Syntax

Daten1 [NOT] LIKE Daten2

Parameter

Daten1

Der Zellenbereich, der verglichen werden soll. Dabei kann es sich um eine Zeichenfolge oder um einen Ausdruck handeln, dessen Auswertung eine Zeichenfolge ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Das Textmuster, mit dem alle Werte in der angegebenen Spalte verglichen werden sollen. Dabei kann es sich um eine Zeichenfolge oder um einen Ausdruck handeln, dessen Auswertung eine Zeichenfolge ergibt. Die Anzahl der Spalten in Daten2 muss mit der Anzahl der Spalten in Daten1 übereinstimmen, es sei denn, bei Daten2 handelt es sich um eine Konstante. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Ein Unterstrich () in Daten2 stellt ein Platzhalterzeichen dar, das mit einem beliebigen einzelnen Zeichen in Daten1 übereinstimmt. Ein Prozentzeichen (%) stimmt mit null oder mehr Zeichen in Daten1 überein.

Beschreibung

LIKE vergleicht die beiden angegebenen Datenbereiche und gibt eine Eins zurück, wenn die Zeichenfolgen übereinstimmen, bzw. eine Null, wenn die Zeichenfolgen nicht übereinstimmen. LIKE gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten1 mit der entsprechenden Spalte von Daten2 verglichen wird (d. h., die erste Spalte von Daten1 wird mit der ersten Spalte von Daten2 verglichen, die zweite Spalte mit der zweiten Spalte usw.).

Wenn es sich bei Daten2 um eine Zeichenfolgekonstante handelt, wird jede Zeichenfolge in Daten1 mit der betreffenden Zeichenfolge verglichen. Wenn es sich bei Daten2 um eine Spalte handelt, werden die Berechnungen zeilenweise ausgeführt. Die Zeichenfolge der ersten Zeile von Daten1 wird mit der Zeichenfolge der ersten Zeile von Daten2 verglichen, die Zeichenfolge der zweiten Zeile mit der Zeichenfolge der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zur letzten Zeichenfolge der kürzesten Spalte.

Beim Vergleich von Zeichenfolgen spielt die Groß-/Kleinschreibung keine Rolle (d. h., "Ja", "ja", "JA" und "jA" werden gleich behandelt).

Anmerkung: Das Makro LIKE verfügt über eine negative Version, NOT LIKE. Deren Format ist mit dem von LIKE identisch. NOT LIKE gibt eine Eins zurück, wenn die Zeichenfolge in Daten1 nicht mit der durch Daten2 definierten Schablone übereinstimmt.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = "Gold" LIKE "Gold"
Erstellt eine neue Spalte TEMP, die den Wert eins enthält (die beiden Zeichenfolgen stimmen überein).
TEMP = "Nein" LIKE "NEIN"
Erstellt eine neue Spalte TEMP, die den Wert eins enthält (beim Vergleich von Zeichenfolgen wird die Groß-/Kleinschreibung nicht beachtet).
TEMP = W1 LIKE "Gold%"
Erstellt eine neue Spalte TEMP, in der jeder Wert eins ist, wenn der entsprechende Zeilenwert von Spalte W1 mit der Zeichenfolge "Gold", auf die eine beliebige Anzahl von Zeichen folgt, übereinstimmt. Andernfalls ist jeder Wert null.
TEMP = W1 LIKE "G_ld"
Erstellt eine neue Spalte TEMP, in der jeder Wert eins ist, wenn der entsprechende Zeilenwert von Spalte W1 mit der Zeichenfolge aus "G", einem beliebigen Zeichen und den Zeichen "ld" übereinstimmt. Andernfalls ist jeder Wert null.
TEMP = W1 LIKE W1
Erstellt eine neue Spalte TEMP, die nur Einsen enthält (jede Zahl ist gleich sich selbst).
TEMP = W1 LIKE W2
Erstellt eine neue Spalte TEMP, wobei jeder Zeilenwert von Spalte W1 mit dem entsprechenden Zeilenwert von Spalte W2 verglichen wird.
TEMP = W1:W3 LIKE W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Zeichenfolgen von Spalte W1 mit den entsprechenden Zeichenfolgen von Spalte W4 verglichen. Die Spalte WX enthält den Vergleich von Spalte W2 mit Spalte W5. Die Spalte WY enthält den Vergleich von Spalte W3 mit Spalte W6.
TEMP = W1[10:20] LIKE W2 oder TEMP = W1[10:20] LIKE W2[1:11]
Erstellt eine neue Spalte TEMP, deren erste 11 Zellen das Ergebnis des Vergleichs der Zeichenfolgen in Zeile 10-20 von Spalte W1 mit den Zeichenfolgen in Zeile 1-11 von Spalte W2 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
EQ	Gibt TRUE zurück, wenn der eine Datenbereich dem anderen gleich ist.
GE	Gibt TRUE zurück, wenn der eine Datenbereich größer-gleich dem anderen ist.
GT	Gibt TRUE zurück, wenn der eine Datenbereich größer als der andere ist.
LE	Gibt TRUE zurück, wenn der eine Datenbereich kleiner-gleich dem anderen ist.
LT	Gibt TRUE zurück, wenn der eine Datenbereich kleiner als der andere ist.
NE	Gibt TRUE zurück, wenn der eine Datenbereich dem anderen ungleich ist.

LN oder LOG

Syntax

LN(Daten) oder LOG(Daten)

Parameter

Daten

Die numerischen Werte, deren natürlicher Logarithmus berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

LN oder LOG berechnet den natürlichen Logarithmus jedes Wertes im angegebenen Datenbereich. Dabei wird für jede Eingabespalte eine neue Spalte zurückgegeben, die den natürlichen Logarithmus der Zahlen in der entsprechenden Eingabespalte enthält. Natürliche Logarithmen basieren auf der Konstante $e = 2,7182818$. LN ist die Umkehrfunktion der Makrofunktion EXP.

Anmerkung: Alle Werte im angegebenen Datenbereich müssen größer als null sein. Andernfalls wird für jede ungültige Eingabe eine leere Zelle zurückgegeben.

Beispiele

TEMP = LN(3) oder TEMP = LOG(3)

Erstellt eine neue Spalte TEMP, die den Wert 1,099 enthält.

TEMP = LN(W1)

Erstellt eine neue Spalte TEMP, in der jeder Wert den natürlichen Logarithmus des Inhalts von Spalte W1 darstellt.

TEMP = LN(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der natürliche Logarithmus des Inhalts von Spalte W1, die Werte von Spalte WX sind der natürliche Logarithmus des Inhalts von Spalte W2 und die Werte von Spalte WY sind der natürliche Logarithmus des Inhalts von Spalte W3.
TEMP = LN(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils den natürlichen Logarithmus der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = LN(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der natürliche Logarithmus der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind der natürliche Logarithmus der entsprechenden Zeilen von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
EXP	Erhebt die natürliche Zahl (e) in die Potenz, die durch den Inhalt jeder Zelle im angegebenen Datenbereich angegeben ist.
LOG2	Berechnet den binären Logarithmus des Inhalts des angegebenen Datenbereichs.
LOG10	Berechnet den dekadischen Logarithmus des Inhalts des angegebenen Datenbereichs.
POW	Erhebt einen Basiswert in die Potenz des oder der angegebenen Exponenten.

LOG2

Syntax

LOG2(Daten)

Parameter

Daten

Die numerischen Werte, deren binärer Logarithmus berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

LOG2 berechnet den binären Logarithmus der Werte im angegebenen Datenbereich. Dabei wird für jede Eingabespalte eine neue Spalte zurückgegeben, die den binären Logarithmus der Zahlen in der entsprechenden Eingabespalte enthält.

Anmerkung: Alle Werte im angegebenen Datenbereich müssen größer als null sein. Andernfalls wird für jede ungültige Eingabe eine leere Zelle zurückgegeben.

Beispiele

TEMP = LOG2(8)
Erstellt eine neue Spalte TEMP, die den Wert drei enthält.
TEMP = LOG2(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert den binären Logarithmus des Inhalts von Spalte W1 darstellt.
TEMP = LOG2(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der binäre Logarithmus des Inhalts von Spalte W1, die Werte von Spalte WX sind der binäre Logarithmus des Inhalts von Spalte W2 und die Werte von Spalte WY sind der binäre Logarithmus des Inhalts von Spalte W3.
TEMP = LOG2(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils den binären Logarithmus der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = LOG2(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der binäre Logarithmus der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind der binäre Logarithmus der entsprechenden Zeilen von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
LN oder LOG	Berechnet den natürlichen Logarithmus des Inhalts des angegebenen Datenbereichs.
LOG10	Berechnet den dekadischen Logarithmus des Inhalts des angegebenen Datenbereichs.
POW	Erhebt einen Basiswert in die Potenz des angegebenen Exponenten.

LOG10

Syntax

LOG10(Daten)

Parameter

Daten

Die numerischen Werte, deren dekadischer Logarithmus berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

LOG10 berechnet den dekadischen Logarithmus der Werte im angegebenen Datenbereich. Dabei wird für jede Eingabespalte eine neue Spalte zurückgegeben, die den dekadischen Logarithmus der Zahlen in der entsprechenden Eingabespalte enthält.

Anmerkung: Alle Werte im angegebenen Datenbereich müssen größer als null sein. Andernfalls wird für jede ungültige Eingabe eine leere Zelle zurückgegeben.

Beispiele

TEMP = LOG10(100)
Erstellt eine neue Spalte TEMP, die den Wert zwei enthält.
TEMP = LOG10(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert den dekadischen Logarithmus des Inhalts von Spalte W1 darstellt.
TEMP = LOG10(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der dekadische Logarithmus des Inhalts von Spalte W1, die Werte von Spalte WX sind der dekadische Logarithmus des Inhalts von Spalte W2 und die Werte von Spalte WY sind der dekadische Logarithmus des Inhalts von Spalte W3.
TEMP = LOG10(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils den dekadischen Logarithmus der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.
TEMP = LOG10(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der dekadische Logarithmus der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind der dekadische Logarithmus der entsprechenden Zeilen von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
LN oder LOG	Berechnet den natürlichen Logarithmus des Inhalts des angegebenen Datenbereichs.
LOG2	Berechnet den binären Logarithmus des Inhalts des angegebenen Datenbereichs.
POW	Erhebt einen Basiswert in die Potenz des angegebenen Exponenten.

LOWER

Syntax

LOWER(Daten)

Parameter

Daten

Der Zeichenfolgewart, der in Kleinbuchstaben konvertiert werden soll.

Beschreibung

LOWER konvertiert jeden Zeichenfolgewart im angegebenen Datenbereich in Kleinbuchstaben. Dabei wird eine neue Spalte zurückgegeben, in der jede Zelle die in Kleinbuchstaben umgewandelte Zeichenfolge der entsprechenden Eingangszelle enthält.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

Temp = LOWER "GOLD"
Erstellt eine neue Spalte TEMP, die den Wert "gold" enthält.
TEMP = LOWER("15. JAN. 1997")
Erstellt eine neue Spalte TEMP, die die ASCII-Zeichenfolge "15. jan. 1997" enthält.
TEMP = LOWER("Druck")
Erstellt eine neue Spalte TEMP, die die ASCII-Zeichenfolge "druck" enthält.
TEMP = LOWER(W1)
Erstellt eine neue Spalte TEMP, die die Kleinbuchstaben jeder Zeichenfolge in der Spalte W1 enthält.

LT

Syntax

Daten1 LT Daten2; Daten1 < Daten2

Parameter

Daten1

Der numerische Zellenbereich, der verglichen werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Die Zahlen, mit denen alle Werte in der angegebenen Spalte verglichen werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

LT vergleicht die beiden angegebenen Datenbereiche und gibt eine Eins zurück, wenn die Werte im ersten Dataset kleiner als die Werte im zweiten Dataset sind, andernfalls eine Null. LT gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten1 mit der entsprechenden Spalte von Daten2 verglichen wird (d. h., die erste Spalte von Daten1 wird mit der ersten Spalte von Daten2 verglichen, die zweite Spalte mit der zweiten Spalte usw.).

Wenn es sich bei Daten2 um eine Konstante handelt, wird jeder Wert in Daten1 mit dem betreffenden Wert verglichen. Wenn es sich bei Daten2 um eine Spalte handelt, werden die Berechnungen zeilenweise ausgeführt. Die erste Zeile von Daten1 wird mit der ersten Zeile von Daten2 verglichen, die zweite Zeile mit der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Der Operator LT kann durch ein Kleiner-als-Zeichen abgekürzt werden (<).

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = 3 LT 4 oder TEMP = 3 < 4
Erstellt eine neue Spalte TEMP, die den Wert eins enthält (drei ist kleiner als vier).
TEMP = W1 < 8
Erstellt eine neue Spalte TEMP, in der jeder Wert eins ist, wenn der entsprechende Zeilenwert von Spalte W1 kleiner als die Zahl acht ist; andernfalls ist der Wert null.
TEMP = W1:W3 < 2
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind der mit zwei verglichene Inhalt von Spalte W1, die Werte von Spalte WX sind der mit zwei verglichene Inhalt von Spalte W2 und die Werte von Spalte WY sind der mit zwei verglichene Inhalt von Spalte W3.
TEMP = W1 < W1
Erstellt eine neue Spalte TEMP, die nur Nullen enthält (keine Zahl ist kleiner als sie selbst).
TEMP = W1 < W2
Erstellt eine neue Spalte TEMP, wobei jeder Zeilenwert von Spalte W1 mit dem entsprechenden Zeilenwert von Spalte W2 verglichen wird.
TEMP = W1[10:20] < W2 oder TEMP = W1[10:20] < W2[1:11]
Erstellt eine neue Spalte TEMP, deren erste 11 Zellen das Ergebnis des Vergleichs der Werte in Zeile 10-20 von Spalte W1 mit den Werten in Zeile 1-11 von Spalte W2 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
EQ	Gibt TRUE zurück, wenn der eine Datenbereich dem anderen gleich ist.
GE	Gibt TRUE zurück, wenn der eine Datenbereich größer-gleich dem anderen ist.

Funktion	Beschreibung
GT	Gibt TRUE zurück, wenn der eine Datenbereich größer als der andere ist.
LE	Gibt TRUE zurück, wenn der eine Datenbereich kleiner-gleich dem anderen ist.

LTRIM

Syntax

LTRIM(Daten)

Parameter

Daten

Die Zeichenfolge, aus der führende Leerzeichen entfernt werden sollen.

Beschreibung

LTRIM entfernt führende Leerzeichen aus jedem Zeichenfolgewert im angegebenen Datenbereich und gibt die konvertierte Zeichenfolge zurück. LTRIM gibt für jede Eingabespalte eine neue Spalte zurück.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

```
Temp = LTRIM " Gold"
```

Erstellt eine neue Zeichenfolge Temp, die "Gold" enthält.

MAX

Syntax

MAX(Daten [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, deren Maximum berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabedatenbereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Anmerkung: Viele Makrofunktionen setzen die Schlüsselwortparameter {ALL | COL | ROW} ein. Diese Schlüsselwörter gelten nicht für IBM Kampagne, da es sich bei den Eingabedaten immer um eine Einzelspalte oder ein Einzelfeld handelt. Das Makro verhält sich immer so, als ob das Schlüsselwort COL angegeben würde. Deshalb brauchen Sie diese Schlüsselwörter nicht anzugeben, wenn Sie IBM Kampagne verwenden.

Beschreibung

MAX berechnet das Maximum der Werte im angegebenen Datenbereich. Es wird eine einzelne neue Spalte zurückgegeben, die den Maximalwert enthält.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = MAX(3) oder TEMP = MAX(3, ALL)
Erstellt eine neue Spalte TEMP, die den Wert drei enthält.
TEMP = MAX(SELECT(COLUMN(1,3,5), W1:W5))
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das Maximum des Inhalts der Spalten W1, W3 und W5 darstellt.
TEMP = MAX(W1)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der den Maximalwert des Inhalts von Spalte W1 darstellt.
TEMP = MAX(W1:W3)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das Maximum des Inhalts der Spalten W1, W2 und W3 darstellt.
TEMP = MAX(W1[10:20])
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das Maximum der Zellen in Zeile 10-20 von Spalte W1 darstellt.
TEMP = MAX(W1[1:5]:W4)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das Maximum der Zellen in Zeile 1-5 von Spalte W1 bis W4 darstellt.
TEMP = MAX(W1:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY. Der Einzelwert in der Spalte TEMP ist das Maximum des Inhalts von Spalte W1, der Einzelwert in der Spalte WX ist das Maximum des Inhalts von Spalte W2 und der Einzelwert in der Spalte WY ist das Maximum des Inhalts von Spalte W3.

TEMP = MAX(W1[1:5]:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils einen Einzelwert enthalten. Der Wert in der Spalte TEMP ist das Maximum der Zellen in Zeile 1-5 von Spalte W1, der Wert in der Spalte WX ist das Maximum der Zellen in Zeile 1-5 von Spalte W2 und der Wert in der Spalte WY ist das Maximum der Zellen in Zeile 1-5 von Spalte W3.
TEMP = MAX(W1:W3, ROW)
Erstellt eine neue Spalte TEMP, in der jeder Zelleneintrag das Maximum der entsprechenden Zeile in Spalte W1, W2 und W3 darstellt.
TEMP = MAX(W1[10:20]:W3, ROW)
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen das Maximum der Werte in Zeile 10-20 von Spalte W1 bis W3 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
DECIMATE	Wandelt eine Spalte von Zahlen in mehrere Spalten um, bei denen eine Eins den Indexwert angibt.
MAXINDEX	Gibt den Spaltenindex des n-ten (ersten, zweiten, dritten usw.) Maximalwerts für jede Zeile der angegebenen Spalte zurück.
MIN	Berechnet das Minimum eines Zellenbereichs.

MAXINDEX



Syntax

MAXINDEX(Daten [, n])

Parameter

Daten

Der Anfang eines Datenbereichs, für den der Index des n-ten Maximalwerts für jede Zeile berechnet werden soll. Dabei kann es sich um eine Spalte oder um einen Ausdruck handeln, dessen Auswertung eine Spalte ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

n

Eine Zahl größer als null. Der Standardwert ist eins; dadurch wird der Index des höchsten Wertes für jede Zeile zurückgegeben. (Bei einer Zwei wird der Index des zweithöchsten Wertes, bei einer Drei der Index des dritthöchsten Wertes zurückgegeben usw.)

Beschreibung

MAXINDEX findet den n-ten Maximalwert in jeder Zeile des angegebenen Datenbereichs und gibt einen Spaltenindex für dessen Position zurück. MAXINDEX gibt eine neue Spalte mit einem Einzelwert zurück, der den Index des n-ten Maximalwerts für jede Zeile darstellt. Der Wert eins steht für die erste Zelle in der ersten Spalte. Wenn es mehrere Maximalwerte gibt, wird die erste Spalte zurückgegeben, die das n-te Maximum enthält.

Anmerkung: Wenn mehrere Spalten vorliegen, die jeweils eine separate Ausgabe-Klasse darstellen, können Sie mit MAXINDEX die "Gewinnerklasse" auswählen. Anschließend können Sie das Training mit einer einzelnen Ausgabespalte statt mit mehreren Ausgabespalten durchführen. Beispielsweise würde $W4 = \text{MAXINDEX}(W1:W3)$ Folgendes ergeben:

W1 W2 W3 W4 0 1 0 2 1 0 0 1 0 0 1 3

Beispiele

TEMP = MAXINDEX(MERGE(3,5,-2))
Erstellt eine neue Spalte TEMP, die den Wert zwei enthält (da der Maximalwert fünf ist, der in der zweiten Spalte auftritt).
TEMP = MAXINDEX(W1)
Erstellt eine neue Spalte TEMP, die für jede Zeile von Spalte W1 eine Eins enthält.
TEMP=MAXINDEX (W6:W8,3)
Erstellt eine neue Spalte TEMP, in der jeder Wert den Index des Minimalwerts (in diesem Fall des dritthöchsten Werts von 3) der entsprechenden Zeile in den Spalten W6, W7 und W8 darstellt. Eine Eins wird zurückgegeben, wenn das Minimum in Spalte W6 vorliegt, eine Zwei, wenn es in Spalte W7 vorliegt, und eine Drei, wenn es in Spalte W8 vorliegt.
TEMP = MAXINDEX(W6:W8)
Erstellt eine neue Spalte TEMP, in der jeder Wert den Index des Maximalwerts der entsprechenden Zeile in Spalte W6, W7 und W8 darstellt. Eine Eins wird zurückgegeben, wenn das Maximum in Spalte W6 vorliegt, eine Zwei, wenn es in Spalte W7 vorliegt, und eine Drei, wenn es in Spalte W8 vorliegt.
TEMP = MAXINDEX(W1[1:5]:W3]
Erstellt eine neue Spalte TEMP, in der die Zellen in Zeile 1-5 den Index des Maximalwerts der entsprechenden Zeile in Spalte W1 bis W3 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
DECIMATE	Wandelt eine Spalte von Zahlen in mehrere Spalten um, bei denen eine Eins den Indexwert angibt.
MAX	Berechnet das Maximum eines Zellenbereichs.
MIN	Berechnet das Minimum eines Zellenbereichs.

Syntax

MEAN(Daten [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, deren arithmetisches Mittel berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabedatenbereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern finden Sie unter „DATE“ auf Seite 77.

Anmerkung: Viele Makrofunktionen setzen die Schlüsselwortparameter {ALL | COL | ROW} ein. Diese Schlüsselwörter gelten nicht für **IBM Kampagne**, da es sich bei den Eingabedaten immer um eine Einzelspalte oder ein Einzelfeld handelt. Das Makro verhält sich immer so, als ob das Schlüsselwort COL angegeben würde. Deshalb brauchen Sie diese Schlüsselwörter nicht anzugeben, wenn Sie **IBM Kampagne** verwenden.

Beschreibung

MEAN berechnet das arithmetische Mittel oder den Durchschnitt der Zellen im angegebenen Datenbereich. Das arithmetische Mittel wird berechnet, indem der Inhalt aller Zellen addiert und durch die Anzahl der Zellen geteilt wird. Die Anzahl der von MEAN zurückgegebenen Spalten hängt von Schlüsselwort ab.

- Wenn Schlüsselwort den Wert ALL hat, gibt MEAN eine neue Spalte zurück, die einen Einzelwert (den Durchschnitt aller Zellen in Daten) enthält.
- Wenn Schlüsselwort den Wert COL hat, gibt MEAN für jede Eingabespalte eine neue Spalte zurück. Jede neue Spalte enthält einen Wert (den Durchschnitt aller Zellen in der entsprechenden Eingabespalte).
- Wenn Schlüsselwort den Wert ROW hat, gibt MEAN eine neue Spalte zurück, die den Durchschnitt für jede Zeile von Daten enthält.

Anmerkung: Leere Zellen werden bei der Berechnung des Durchschnitts ignoriert.

Anmerkung: MEAN ist mit der Makrofunktion AVG identisch.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

<pre>TEMP = MEAN(MERGE(3, 4, 5)) oder TEMP = MEAN(MERGE(3, 4, 5), ALL)</pre>
Erstellt eine neue Spalte TEMP, die den Wert 4 enthält.
<pre>TEMP = MEAN(MERGE(-10, 6, 10))</pre>
Erstellt eine neue Spalte TEMP, die den Wert 2 enthält.
<pre>TEMP = MEAN(W1)</pre>
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das arithmetische Mittel des Inhalts von Spalte W1 darstellt.
<pre>TEMP = MEAN(W1:W3)</pre>
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das arithmetische Mittel des Inhalts der Spalten W1, W2 und W3 darstellt.
<pre>TEMP = MEAN(W1[10:20])</pre>
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das arithmetische Mittel der Zellen in Zeile 10-20 von Spalte W1 darstellt.
<pre>TEMP = MEAN(W1[1:5]:W4)</pre>
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das arithmetische Mittel der Zellen in Zeile 1-5 von Spalte W1 bis W4 darstellt.
<pre>TEMP = MEAN(W1:W3, COL)</pre>
Erstellt drei neue Spalten TEMP, WX und WY. Der Einzelwert in der Spalte TEMP ist das arithmetische Mittel des Inhalts von Spalte W1, der Einzelwert in der Spalte WX ist das arithmetische Mittel des Inhalts von Spalte W2 und der Einzelwert in der Spalte WY ist das arithmetische Mittel des Inhalts von Spalte W3.
<pre>TEMP = MEAN(MERGE(1,4), COL)</pre>
Erstellt zwei neue Spalten TEMP und WX. TEMP enthält den Einzelwert eins; WX enthält den Einzelwert vier.
<pre>TEMP = MEAN(W1[10:20]:W3, COL)</pre>
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils einen Einzelwert enthalten. Der Wert in der Spalte TEMP ist das arithmetische Mittel der Zellen in Zeile 10-20 von Spalte W1, der Wert in der Spalte WX ist das arithmetische Mittel der Zellen in Zeile 10-20 von Spalte W2 und der Wert in der Spalte WY ist das arithmetische Mittel der Zellen in Zeile 10-20 von Spalte W3.
<pre>TEMP = MEAN(W1:W3, ROW)</pre>
Erstellt eine neue Spalte TEMP, in der jeder Zelleneintrag das arithmetische Mittel der entsprechenden Zeile in Spalte W1, W2 und W3 darstellt.
<pre>TEMP = MEAN(W1[1:5]:W3, ROW)</pre>
Erstellt eine neue Spalte TEMP, in der die Zellen in Zeile 1-5 das arithmetische Mittel der entsprechenden Zeile in Spalte W1 bis W3 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
AVG_DEV	Berechnet die mittlere Abweichung eines Zellenbereichs.
SUM oder TOTAL	Berechnet die Summe eines Zellenbereichs.

MERGE



Syntax

MERGE(Daten [, Daten]...) {Daten [, Daten]...}

Parameter

Daten

Der Name einer Spalte, die zu einem Datenbereich kombiniert werden soll. Dabei kann es sich um einen konstanten Wert (numerisch oder ASCII-Text in Anführungszeichen), eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Dieser Parameter kann mehrmals wiederholt werden. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

MERGE verkettet die Eingabe horizontal zu einer neuen Gruppe. MERGE gibt so viele neue Spalten zurück, wie in der Eingabe enthalten sind. Es kann eine unbegrenzte Zahl von Argumenten angegeben werden.

Anmerkung: Die Makrofunktion MERGE kann durch geschweifte Klammern ({}), abgekürzt werden. Die Argumente können einfach durch Kommas getrennt in geschweifte Klammern eingefügt werden (z. B. ist TEMP = {1,2,3} äquivalent zu TEMP = MERGE(1,2,3)).

Beispiele

TEMP = MERGE(3, 4, "fünf") oder TEMP = {3, 4, "fünf"}
Erstellt drei neue Spalten TEMP, WX und WY, die den Wert 3, 4 bzw. "fünf" enthalten.
TEMP = W1:W3 oder TEMP = MERGE(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Dabei ist TEMP eine Kopie von Spalte W1, WX eine Kopie von Spalte W2 und WY eine Kopie von Spalte W3.
TEMP = MERGE(W1, W3, W5:W7)
Erstellt fünf neue Spalten TEMP, WW, WX, WY und WZ. TEMP ist eine Kopie von Spalte W1; WW ist eine Kopie von Spalte W3; WX bis WZ sind Kopien von Spalte W5 bis W7.

TEMP = AVG(MERGE(W1,W3,W5), ROW)
Erstellt eine neue Spalte TEMP, in der jede Zeile den Durchschnitt der entsprechenden Zeile in Spalte W1, W3 und W5 darstellt. Ein Durchschnitt wird nur bis zur letzten Zeile der kürzesten Eingabespalte berechnet.
TEMP = MERGE(W1[10:50],W3, W5:W7[1:30])
Erstellt fünf neue Spalten TEMP, WW, WX, WY und WZ. TEMP ist eine Kopie der Werte in Zeile 10-50 von Spalte W1; WW ist eine Kopie der Werte von Spalte W3; WX bis WZ sind Kopien der Werte in Zeile 1-30 von Spalte W5 bis W7.
TEMP = AVG(MERGE(W1, W5:W6)) oder TEMP = AVG({W1, W5:W6})
Erstellt eine neue Spalte TEMP, die den Durchschnitt aller Zellen in Spalte W1, W5 und W6 enthält.

Zugehörige Funktionen

Funktion	Beschreibung
COLUMN	Erstellt neue Spalten, wobei die Eingabewerte in jeder Spalte vertikal verketten werden.
SELECT	Gibt die angegebene(n) Spalte(n) aus einem Datenbereich zurück.
TRANPOSE	Setzt einen angegebenen Datenbereich um.

MIN

Syntax

MIN(Daten [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, deren Minimum berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabedatenbereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern finden Sie unter „DATE“ auf Seite 77.

Anmerkung: Viele Makrofunktionen setzen die Schlüsselwortparameter {ALL | COL | ROW} ein. Diese Schlüsselwörter gelten nicht für **IBM Kampagne**, da es sich bei den Eingabedaten immer um eine Einzelspalte oder ein Einzelfeld handelt. Das Makro verhält sich immer so, als ob das Schlüsselwort COL angegeben würde. Deshalb brauchen Sie diese Schlüsselwörter nicht anzugeben, wenn Sie **IBM Kampagne** verwenden.

Beschreibung

MIN berechnet das Minimum aller Zellen im angegebenen Datenbereich. Es wird eine einzelne neue Spalte zurückgegeben, die den Minimalwert enthält.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = MIN(MERGE(1,10,-2))	Erstellt eine neue Spalte TEMP, die den Wert -2 enthält.
TEMP = MIN(W1)	Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der den Minimalwert des Inhalts von Spalte W1 darstellt.
TEMP = MIN(W1:W3)	Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das Minimum des Inhalts der Spalten W1, W2 und W3 darstellt.
TEMP = MIN(W1[10:20])	Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das Minimum der Zellen in Zeile 10-20 von Spalte W1 darstellt.
TEMP = MIN(W1[1:5]:W4)	Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der das Minimum der Zellen in Zeile 1-5 von Spalte W1 bis W4 darstellt.
TEMP = MIN(W1:W3, COL)	Erstellt drei neue Spalten TEMP, WX und WY. Der Einzelwert in der Spalte TEMP ist das Minimum des Inhalts von Spalte W1, der Einzelwert in der Spalte WX ist das Minimum des Inhalts von Spalte W2 und der Einzelwert in der Spalte WY ist das Minimum des Inhalts von Spalte W3.
TEMP = MIN(W1[1:5]:W3, COL)	Erstellt drei neue Spalten TEMP, WX und WY, die jeweils einen Einzelwert enthalten. Der Wert in der Spalte TEMP ist das Minimum der Zellen in Zeile 1-5 von Spalte W1, der Wert in der Spalte WX ist das Minimum der Zellen in Zeile 1-5 von Spalte W2 und der Wert in der Spalte WY ist das Minimum der Zellen in Zeile 1-5 von Spalte W3.
TEMP = MIN(W1:W3, ROW)	Erstellt eine neue Spalte TEMP, in der jeder Zelleneintrag das Minimum der entsprechenden Zeile in Spalte W1, W2 und W3 darstellt.
TEMP = MIN(W1[10:20]:W3, ROW)	Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen das Minimum der Werte in Zeile 1-5 von Spalte W1 bis W3 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
DECIMATE	Wandelt eine Spalte von Zahlen in mehrere Spalten um, bei denen eine Eins den Indexwert angibt.
MAX	Berechnet das Maximum eines Zellenbereichs.
MAX_TO_INDEX	Gibt für jede Zeile der angegebenen Spalte den Spaltenindex des Maximalwerts zurück.

MINUS

Syntax

Daten MINUS Subtrahend; Daten - Subtrahend

Parameter

Daten

Der Zellenbereich mit Zahlen, von denen etwas subtrahiert werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Subtrahend

Die Zahl(en), die von allen Werten in der angegebenen Spalte subtrahiert werden soll(en). Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Subtrahend muss mit der Anzahl der Spalten in Daten übereinstimmen, es sei denn, bei Subtrahend handelt es sich um eine Konstante. Informationen zur Formatdefinition von Subtrahend (identisch mit Daten) finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

MINUS subtrahiert Subtrahend vom angegebenen Datenbereich Daten. MINUS gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Subtrahend von der entsprechenden Spalte von Daten subtrahiert wird (d. h., von der ersten Spalte von Daten wird die erste Spalte von Subtrahend subtrahiert, von der zweiten Spalte die zweite Spalte usw.).

Wenn es sich bei Subtrahend um eine Konstante handelt, wird der betreffende Wert von jedem Wert in Daten subtrahiert. Wenn Subtrahend eine oder mehrere Spalten enthält, werden die Berechnungen zeilenweise zwischen einer Spalte von Daten und einer Spalte von Subtrahend ausgeführt. Von der ersten Zeile von Daten wird der erste Zeilenwert von Subtrahend subtrahiert, von der zweiten Zeile der zweite Zeilenwert usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Der Operator MINUS kann durch ein Minuszeichen oder einen Bindestrich (-) abgekürzt werden.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = 7 MINUS 4 oder TEMP = 7 - 4
Erstellt eine neue Spalte TEMP, die den Wert drei enthält.
TEMP = W1 - 8
Erstellt eine neue Spalte TEMP, in der jeder Wert den Inhalt von Spalte W1 minus acht darstellt.
TEMP = W1:W3 - 2
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind der Inhalt von Spalte W1 minus zwei, die Werte von Spalte WX sind der Inhalt von Spalte W2 minus zwei und die Werte von Spalte WY sind der Inhalt von Spalte W3 minus zwei.
TEMP = W1 - W1
Erstellt eine neue Spalte TEMP, die nur Nullen enthält (jede Zahl minus dieselbe Zahl ergibt null).
TEMP = W1 - W2
Erstellt eine neue Spalte TEMP, wobei jeder Wert den Zeilenwert von Spalte W1 minus den entsprechenden Zeilenwert von Spalte W2 darstellt.
TEMP = W1:W3 - W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. Die Spalte TEMP enthält die Werte von Spalte W1 minus die entsprechenden Zeilenwerte von Spalte W4. Die Spalte WX subtrahiert Spalte W5 von Spalte W2. Die Spalte WY subtrahiert Spalte W6 von Spalte W3.
TEMP = W1[10:20] - W2 oder TEMP = W1[10:20] - W2[1:11]
Erstellt eine neue Spalte TEMP, deren erste 11 Zellen die Werte in Zeile 10-20 von Spalte W1 minus die Werte in Zeile 1-11 von Spalte W2 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
PLUS	Addiert die Inhalte zweier Datenbereiche.
SUM oder TOTAL	Berechnet die Summe eines Zellenbereichs.

MOD

Syntax

Daten MOD Divisor; Daten % Divisor

Parameter

Daten

Die Ganzzahlwerte, deren Modulo-Wert berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck

handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Divisor

Die Ganzzahl ungleich null, bezüglich deren der Modulo-Wert berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Divisor muss mit der Anzahl der Spalten in Daten übereinstimmen, es sei denn, bei Divisor handelt es sich um eine Konstante. Informationen zur Formatdefinition von Divisor (identisch mit Daten) finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

MOD berechnet den Rest der Division des angegebenen Datenbereichs durch einen angegebenen Wert. Dieser wird berechnet, indem jeder Wert durch Divisor geteilt und der Rest zurückgegeben wird. MOD gibt für jede Eingabespalte eine neue Spalte zurück, die die Zahlen in Daten modulo Divisor enthält. Der Rest hat dasselbe Vorzeichen (positiv oder negativ) wie Daten.

Wenn es sich bei Divisor um eine Konstante handelt, wird für jeden Wert in der angegebenen Spalte der betreffende Modulo-Wert berechnet. Wenn es sich bei Divisor um eine Spalte handelt, werden die Berechnungen zeilenweise ausgeführt. Bei der Modulo-Berechnung wird die erste Zeile in Daten dem ersten Zeilenwert von Divisor zugeordnet, die zweite Zeile dem zweiten Zeilenwert usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Wenn Divisor null ist, wird der Fehler 'Division durch null' zurückgegeben.

Anmerkung: Der Operator MOD kann durch ein Prozentzeichen (%) abgekürzt werden. Beispielsweise ist $TEMP = 5 \% 3$ äquivalent zu $TEMP = 5 MOD 3$.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = 10 MOD 8 oder TEMP = 10 % 8
Erstellt eine neue Spalte TEMP, die den Wert 2 enthält.
TEMP = -10 % 8
Erstellt eine neue Spalte TEMP, die den Wert -2 enthält.
TEMP = W1 % 8
Erstellt eine neue Spalte TEMP, in der jeder Wert den Inhalt von Spalte W1 modulo acht darstellt.

TEMP = W1:W3 % 2
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die Werte modulo zwei des Inhalts von Spalte W1, die Werte von Spalte WX sind die Werte modulo zwei des Inhalts von Spalte W2 und die Werte von Spalte WY sind die Werte modulo zwei des Inhalts von Spalte W3.
TEMP = W1 % W1
Erstellt eine neue Spalte TEMP, die für jeden Eintrag in Spalte W1 eine Null enthält. Jede Zahl modulo die Zahl selbst ergibt null.
TEMP = W1 % W2
Erstellt eine neue Spalte TEMP, wobei jeder Wert den Zeilenwert von Spalte W1 modulo den entsprechenden Zeilenwert von Spalte W2 darstellt. Wenn W2=W1, werden nur Nullen zurückgegeben, wie im vorherigen Beispiel.
TEMP = W1:W3 % W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. Die Spalte TEMP enthält die Werte von Spalte W1 modulo die entsprechenden Zeilenwerte von Spalte W4. Die Spalte WX enthält die Ergebnisse von Spalte W2 modulo W5. Die Spalte WY enthält die Ergebnisse von Spalte W3 modulo W6.
TEMP = W1[10:20] % W2 oder TEMP = W1[10:20] % W2[1:11]
Erstellt eine neue Spalte TEMP, deren erste 11 Zellen die Werte in Zeile 10-20 von Spalte W1 modulo die Werte in Zeile 1-11 von Spalte W2 darstellen. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
DIV	Dividiert einen angegebenen Datenbereich durch einen anderen.
MOD	Berechnet den Modulo-Wert des Inhalts des angegebenen Datenbereichs.

MONTHOF

Syntax

MONTHOF(Datumszeichenfolge [, Eingabeformat])

Parameter

Datumszeichenfolge

Ein Text, der ein gültiges Datum darstellt.

Eingabeformat

Eines der Schlüsselwörter in der folgenden Tabelle, das das Datumsformat von Datumszeichenfolge angibt.

Beschreibung

MONTHOF gibt für das durch Datumszeichenfolge angegebene Datum den Monat als Zahl zurück. Wenn Eingabeformat nicht angegeben ist, wird das Standardschlüsselwort BEGRENZ_M_T_J verwendet.

Beispiele

MONTHOF("012171", MMTTJJ) gibt die Zahl 1 zurück.

Anmerkung: Weitere Informationen zu gültigen Datumsformaten finden Sie unter „DATE“ auf Seite 77.

Zugehörige Funktionen

Funktion	Beschreibung
DAYOF	Gibt den Wochentag als Zahl zurück.
WEEKDAYOF	Gibt den Wochentag der Woche als Zahl zurück.
YEAROF	Gibt das Jahr als Zahl zurück.

MULT

Syntax

Daten MULT Multiplikator; Daten * Multiplikator

Parameter

Daten

Die numerischen Werte, die multipliziert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Multiplikator

Die Zahl, mit der alle Werte in der angegebenen Spalte multipliziert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Multiplikator muss mit der Anzahl der Spalten in Daten übereinstimmen, es sei denn, bei Multiplikator handelt es sich um eine Konstante. Informationen zur Formatdefinition von Multiplikator (identisch mit Daten) finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

MULT multipliziert die Werte in den beiden angegebenen Datenbereichen. MULT gibt für jede Eingabespalte eine neue Spalte zurück, die die mit Multiplikator multiplizierten Zahlen in Daten enthält. Wenn es sich bei Multiplikator um eine Konstante handelt, wird jeder Wert in Daten mit dem betreffenden Wert multipliziert. Wenn

es sich bei Multiplikator um eine Spalte handelt, werden die Berechnungen zeilenweise ausgeführt. Die erste Zeile von Daten wird mit der ersten Zeile von Multiplikator multipliziert, die zweite Zeile mit der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Wenn eine Spalte in jeder Zeile dieselbe Zahl x wie Multiplikator enthält, ist dies dasselbe, als wenn als Multiplikator die Konstante x verwendet wird.

Anmerkung: Der Operator MULT kann durch einen Stern (*) abgekürzt werden.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = 8 MULT 4 oder TEMP = 8 * 4
Erstellt eine neue Spalte TEMP, die den Wert 32 enthält.
TEMP = W1 * 8
Erstellt eine neue Spalte TEMP, in der jeder Wert den Inhalt von Spalte W1 darstellt, der mit acht multipliziert ist.
TEMP = W1:W3 * 2
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils das Zweifache des Inhalts von Spalte W1, die Werte von Spalte WX sind das Zweifache des Inhalts von Spalte W2 und die Werte von Spalte WY sind das Zweifache des Inhalts von Spalte W3.
TEMP = W1 * W1
Erstellt eine neue Spalte TEMP, die das Quadrat jedes Werts in Spalte W1 enthält.
TEMP = W1 * W2
Erstellt eine neue Spalte TEMP, wobei jeder Zeilenwert von Spalte W1 mit dem entsprechenden Zeilenwert von Spalte W2 multipliziert ist.
TEMP = W1:W3 * W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Werte von Spalte W1 mit den entsprechenden Zeilenwerten von Spalte W4 multipliziert. Die Spalte WX enthält die Multiplikation von Spalte W2 mit Spalte W5. Die Spalte WY enthält die Multiplikation von Spalte W3 mit Spalte W6.
TEMP = W1[10:20] * W2 oder TEMP = W1[10:20] * W2[1:11]
Erstellt eine neue Spalte TEMP, deren erste 11 Zellen das Ergebnis der Multiplikation der Werte in Zeile 10-20 von Spalte W1 mit den Werten in Zeile 1-11 von Spalte W2 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
DIV	Dividiert einen angegebenen Datenbereich durch einen anderen.
EXP	Erhebt die natürliche Zahl (e) in die Potenz, die durch den Inhalt jeder Zelle im angegebenen Datenbereich angegeben ist.

Funktion	Beschreibung
POW	Erhebt einen Basiswert in die Potenz des oder der angegebenen Exponenten.

NE

Syntax

Daten1 NE Daten2; Daten1 != Daten2; Daten1 <> Daten2

Parameter

Daten1

Der Zellenbereich, der verglichen werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Die Zahlen, mit denen alle Werte in der angegebenen Spalte verglichen werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Daten2 muss mit der Anzahl der Spalten in Daten1 übereinstimmen, es sei denn, bei Daten2 handelt es sich um eine Konstante. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

NE vergleicht die beiden angegebenen Datenbereiche und gibt eine Eins zurück, wenn die Werte ungleich sind, bzw. eine Null, wenn die Werte gleich sind. NE gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten1 mit der entsprechenden Spalte von Daten2 verglichen wird (d. h., die erste Spalte von Daten1 wird mit der ersten Spalte von Daten2 verglichen, die zweite Spalte mit der zweiten Spalte usw.).

Wenn es sich bei Daten2 um eine Konstante handelt, wird jeder Wert in Daten1 mit dem betreffenden Wert verglichen. Wenn es sich bei Daten2 um eine Spalte handelt, werden die Berechnungen zeilenweise ausgeführt. Die erste Zeile von Daten1 wird mit der ersten Zeile von Daten2 verglichen, die zweite Zeile mit der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Wenn eine Spalte in jeder Zeile dieselbe Zahl x wie Daten2 enthält, ist dies dasselbe, als wenn als Daten2 die Konstante x verwendet wird.

Anmerkung: Der Operator NE kann durch die Folge aus Ausrufezeichen und Gleichheitszeichen (!=) oder durch die Folge aus Kleiner-als-Zeichen und Größer-als-Zeichen (<>) abgekürzt werden.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = 3 NE 4 oder TEMP = 3 != 4; TEMP = 3 <> 4
Erstellt eine neue Spalte TEMP, die den Wert eins enthält (drei ist ungleich vier).
TEMP = W1 != 8
Erstellt eine neue Spalte TEMP, in der jeder Wert eins ist, wenn der entsprechende Zeilenwert von Spalte W1 ungleich der Zahl acht ist; andernfalls ist der Wert null.
TEMP = W1:W3 != 2
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind der mit zwei verglichene Inhalt von Spalte W1, die Werte von Spalte WX sind der mit zwei verglichene Inhalt von Spalte W2 und die Werte von Spalte WY sind der mit zwei verglichene Inhalt von Spalte W3.
TEMP = W1 != W1
Erstellt eine neue Spalte TEMP, die nur Nullen enthält (jede Zahl ist gleich sich selbst).
TEMP = W1 != W2
Erstellt eine neue Spalte TEMP, wobei jeder Zeilenwert von Spalte W1 mit dem entsprechenden Zeilenwert von Spalte W2 verglichen wird.
TEMP = W1:W3 != W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Werte von Spalte W1 mit den entsprechenden Zeilenwerten von Spalte W4 verglichen. Die Spalte WX enthält den Vergleich von Spalte W2 mit Spalte W5. Die Spalte WY enthält den Vergleich von Spalte W3 mit Spalte W6.
TEMP = W1[10:20] != W2 oder TEMP = W1[10:20] != W2[1:11]
Erstellt eine neue Spalte TEMP, deren erste 11 Zellen das Ergebnis des Vergleichs der Werte in Zeile 10-20 von Spalte W1 mit den Werten in Zeile 1-11 von Spalte W2 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
EQ	Gibt TRUE zurück, wenn der eine Datenbereich dem anderen gleich ist.
GE	Gibt TRUE zurück, wenn der eine Datenbereich größer-gleich dem anderen ist.
GT	Gibt TRUE zurück, wenn der eine Datenbereich größer als der andere ist.
LE	Gibt TRUE zurück, wenn der eine Datenbereich kleiner-gleich dem anderen ist.
LT	Gibt TRUE zurück, wenn der eine Datenbereich kleiner als der andere ist.

NORM_MINMAX



Syntax

NORM_MINMAX(Daten [, Schlüsselwort]); NORM_MINMAX(Daten, Min, Max [, Schlüsselwort]); NORM_MINMAX(Daten, Basisdaten [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, die normalisiert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Min, Max

Diese beiden Parameter geben einen Minimalwert und einen Maximalwert für die Normalisierung an. Es muss sich um Konstanten handeln, außer bei dem Schlüsselwort ROW, wo es sich um Konstanten oder Spalten handeln kann.

Basisdaten

Dieser Parameter gibt einen Datenbereich an, der für die Berechnung des Minimalwerts und des Maximalwerts für die Normalisierung verwendet wird. Die Anzahl der Spalten in Basisdaten muss mit der Anzahl der Spalten in Daten übereinstimmen. Informationen zur Formatdefinition von Basisdaten (identisch mit Daten) finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabebereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

NORM_MINMAX berechnet die normalisierten Werte des angegebenen Datenbereichs. Alle Rückgabewerte liegen zwischen null und eins (einschließlich). Die Normalisierung wird wie folgt ausgeführt:

$$VX[y] = \frac{VX[y] - \min}{\max - \min}$$

Dabei werden *Min* und *Max* wie folgt festgelegt:

- Wenn *Min* und *Max* angegeben sind, werden diese Werte für das Minimum bzw. das Maximum verwendet. Wenn diese Parameter bei dem Schlüsselwort *ROW* angegeben sind, kann es sich bei *Min* und *Max* um Spalten handeln, die für jede Zeile von Daten einen Minimalwert und einen Maximalwert angeben. Wenn es sich bei *Min* und *Max* um Spalten handelt, müssen die Spalten entweder dieselbe Länge wie Daten aufweisen oder skalar sein (d. h., sie müssen einen Einzelwert enthalten, der als Konstante auf alle Werte in der entsprechenden Spalte von Daten angewendet wird).
- Wenn *Basisdaten* angegeben ist, werden der Minimalwert und der Maximalwert dieses Datenbereichs berechnet und für die Normalisierung von Daten verwendet. Die Spalten in *Basisdaten* müssen mindestens zwei Zellenwerte enthalten.
- Wenn keine dieser sich gegenseitig ausschließenden Optionen angegeben ist, werden der Minimalwert und der Maximalwert automatisch aus Daten errechnet.

Anmerkung: Da alle Rückgabewerte zwischen 0,0 und 1,0 liegen, werden alle Werte, deren Berechnung nach der obigen Gleichung einen Wert kleiner als 0,0 ergibt, als 0,0 zurückgegeben. Entsprechend werden alle Werte größer als 1,0 als 1,0 zurückgegeben. *NORM_MINMAX* gibt immer einen Datenbereich zurück, dessen Dimensionen denen des Eingabedatenbereichs entsprechen. Wenn das Schlüsselwort *ALL* angegeben ist, werden Durchschnitt und Standardabweichung für den gesamten Eingabedatenbereich berechnet. Wenn das Schlüsselwort *COL* angegeben ist, werden Durchschnitt und Standardabweichung für jede Eingabespalte berechnet und für die Normalisierung der betreffenden Spalte verwendet. Wenn das Schlüsselwort *ROW* angegeben ist, werden Durchschnitt und Standardabweichung für jede Zeile im angegebenen Datenbereich berechnet und für die Normalisierung der betreffenden Zeile verwendet.

Anmerkung: Wenn Minimalwert und Maximalwert gleich sind, werden nur Nullen zurückgegeben.

Beispiele

```
TEMP = NORM_MINMAX(3)
```

Erstellt eine neue Spalte *TEMP*, die den Wert null enthält.

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

Erstellt eine neue Spalte *TEMP*, die die Werte 0, 0,5 und 1 enthält. (Minimalwert und Maximalwert [3 und 5] werden automatisch aus dem Datenbereich errechnet.)

```
TEMP = NORM_MINMAX(COLUMN(3, 4, 5), 0, 10)
```

Erstellt eine neue Spalte *TEMP*, die die Werte 0,3, 0,4 und 0,5 enthält. (Hier sind Minimalwert und Maximalwert [0 und 10] als Argumente angegeben.)

TEMP = NORM_MINMAX(W1)
Erstellt eine neue Spalte TEMP, die die normalisierten Werte des Inhalts von Spalte W1 enthält. Der Minimalwert und der Maximalwert für die Normalisierung werden aus der Spalte W1 errechnet.
TEMP = NORM_MINMAX(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Minimalwert und der Maximalwert für die Normalisierung werden aus den Spalten W1, W2 und W3 errechnet.
TEMP = NORM_MINMAX(W1[1:5]:W3)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils in Zeile 1-5 Werte enthalten. Die Werte von Spalte TEMP sind die normalisierten Werte der entsprechenden Zeilen von Spalte W1, die Werte von Spalte WX sind die normalisierten Werte der entsprechenden Zeilen von Spalte W2 und die Werte von Spalte WY sind die normalisierten Werte der entsprechenden Zeilen von Spalte W3. Der Minimalwert und der Maximalwert für die Normalisierung werden aus Zeile 1-5 von Spalte W1-W3 errechnet.
TEMP = NORM_MINMAX(W1:W3, W4:W6)
Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Minimalwert und der Maximalwert für die Normalisierung werden aus den Spalten W4, W5 und W6 errechnet.
TEMP = NORM_MINMAX(W1:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Minimalwert und der Maximalwert für die Normalisierung werden für jede Spalte unabhängig errechnet (d. h. jeweils für Spalte W1, Spalte W2 usw. getrennt).
TEMP = NORM_MINMAX(W1[10:50]:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils in den ersten 41 Zeilen Werte enthalten. Die Werte von Spalte TEMP sind die normalisierten Werte von Zeile 10-50 von Spalte W1, die Werte von Spalte WX sind die normalisierten Werte von Zeile 10-50 von Spalte W2 und die Werte von Spalte WY sind die normalisierten Werte von Zeile 10-50 von Spalte W3. Der Minimalwert und der Maximalwert für die Normalisierung werden für jede Spalte unabhängig aus Zeile 10-50 errechnet.
TEMP = NORM_MINMAX(W1:W3, W4:W6, COL)
Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Minimalwert und der Maximalwert für die Normalisierung werden für jede Spalte unabhängig aus den Spalten W4-W6 errechnet (d. h. aus Spalte W4 für die Normalisierung von Spalte W1, aus Spalte W5 für die Normalisierung von Spalte W2 usw.).
TEMP = NORM_MINMAX(W1:W3, ROW)
Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Minimalwert und der Maximalwert für die Normalisierung werden unabhängig aus jeder Zeile in den Spalten W1, W2 und W3 errechnet.
TEMP = NORM_MINMAX(W1[10:20]:W3, ROW)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils in den ersten 11 Zeilen Werte enthalten. Die Werte von Spalte TEMP sind die normalisierten Werte von Zeile 10-20 von Spalte W1, die Werte von Spalte WX sind die normalisierten Werte von Zeile 10-20 von Spalte W2 und die Werte von Spalte WY sind die normalisierten Werte von Zeile 10-20 von Spalte W3. Der Minimalwert und der Maximalwert für die Normalisierung werden aus Zeile 10-20 von Spalte W1-W3 errechnet.

TEMP = NORM_MINMAX(W1:W3, W8:W10, ROW)

Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Minimalwert und der Maximalwert für die Normalisierung werden unabhängig aus jeder Zeile in den Spalten W8-W10 errechnet.

Zugehörige Funktionen

Funktion	Beschreibung
NORM_SIGMOID	Berechnet die sigmoidale Normalisierung eines Datenbereichs.
NORM_ZSCORE	Berechnet die z-Faktorwert-Normalisierung eines Datenbereichs.

NORM_SIGMOID



Syntax

NORM_SIGMOID(Daten [, Schlüsselwort]); NORM_SIGMOID(Daten, Durchschnitt, Stdabw [, Schlüsselwort]); NORM_SIGMOID(Daten, Basisdaten [, Schlüsselwort])

Parameter

Daten

Die Werte, die normalisiert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Durchschnitt, Stdabw

Diese beiden Parameter geben den Durchschnitt und die Standardabweichung für die Normalisierung an. Es muss sich um Konstanten handeln, außer bei dem Schlüsselwort ROW, wo es sich um Konstanten oder Spalten handeln kann.

Basisdaten

Dieser Parameter gibt einen Datenbereich an, der für die Berechnung des Durchschnitts und der Standardabweichung für die Normalisierung von Daten verwendet wird.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabebereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

NORM_SIGMOID berechnet die normalisierten Werte des angegebenen Datenbereichs. Eine sigmoidale Normalisierung ordnet Daten entlang einer Sigmoidkurve neu an und gibt Werte zwischen -1,0 und +1,0 (einschließlich) zurück. Im Wesentlichen werden alle Daten innerhalb einer Standardabweichung vom Durchschnitt im mittleren Bereich des Sigmoids linear verteilt. Ausreißer werden an den Flanken des Sigmoids dargestellt. Auf diese Weise können stark abweichende Datenpunkte einbezogen werden, ohne dass die Unterscheidbarkeit von Punkten nahe des Durchschnitts beeinträchtigt wird.

Die sigmoidale Normalisierung wird wie folgt ausgeführt:

$$VX[y] = \frac{1 - e^{-\alpha}}{1 + e^{-\alpha}}$$

Dabei werden

$$\alpha = \frac{VX[y] - \text{mean}}{\text{std}}$$

und *Durchschnitt* und *Stdabw* wie folgt festgelegt:

- Wenn Durchschnitt und Stdabw angegeben sind, werden diese Werte für den Durchschnitt bzw. die Standardabweichung verwendet. Wenn diese Parameter bei dem Schlüsselwort ROW angegeben sind, kann es sich bei Durchschnitt und Stdabw um Spalten handeln, die für jede Zeile von Daten einen Durchschnitt und eine Standardabweichung angeben. Wenn es sich bei Durchschnitt und Stdabw um Spalten handelt, müssen die Spalten entweder dieselbe Länge wie Daten aufweisen oder skalar sein (d. h., sie müssen einen Einzelwert enthalten, der als Konstante auf alle Werte in der entsprechenden Spalte von Daten angewendet wird).
- Wenn Basisdaten angegeben ist, werden der Durchschnitt und die Standardabweichung dieses Datenbereichs berechnet und für die Normalisierung von Daten verwendet. Die Spalten in Basisdaten müssen mindestens zwei Zellenwerte enthalten.
- Wenn keine dieser sich gegenseitig ausschließenden Optionen angegeben ist, werden der Durchschnitt und die Standardabweichung automatisch aus Daten errechnet.

NORM_SIGMOID gibt immer einen Datenbereich zurück, dessen Dimensionen denen des Eingabedatenbereichs entsprechen. Wenn das Schlüsselwort ALL angegeben ist, werden Durchschnitt und Standardabweichung für den gesamten Eingabedatenbereich berechnet. Wenn das Schlüsselwort COL angegeben ist, werden Durchschnitt und Standardabweichung für jede Eingabespalte berechnet und für die Normalisierung der betreffenden Spalte verwendet. Wenn das Schlüsselwort ROW angegeben ist, werden Durchschnitt und Standardabweichung für jede Zeile im angegebenen Datenbereich berechnet und für die Normalisierung der betreffenden Zeile verwendet.

Anmerkung: Wenn die Standardabweichung null ist, werden nur Nullen zurückgegeben.

Anmerkung: Um Daten mithilfe desselben Basisdaten-Bereichs (z. B. in eingeschlossenen Benutzerfunktionen) zu normalisieren, müssen Durchschnitt und Stdabw in Konstanten umgewandelt werden (dazu kann die Makrofunktion CONSTANT herangezogen werden).

Beispiele

TEMP = NORM_SIGMOID(COLUMN(3, 4, 5))
Erstellt eine neue Spalte TEMP, die die Werte -0,55, 0 und 0,55 enthält. (Durchschnitt und Standardabweichung [4 und 0,816] werden automatisch aus dem Datenbereich errechnet.)
TEMP = NORM_SIGMOID(COLUMN(3, 4, 5), 3.5, 1.2)
Erstellt eine neue Spalte TEMP, die die Werte -0,21, 0,21 und 0,55 enthält. (Hier sind Durchschnitt und Standardabweichung [3,5 und 1,2] als Argumente angegeben.)
TEMP = NORM_SIGMOID(W1) oder TEMP = NORM_SIGMOID(W1, ALL)
Erstellt eine neue Spalte TEMP, die die normalisierten Werte des Inhalts von Spalte W1 enthält. Der Durchschnitt und die Standardabweichung für die Normalisierung werden aus der Spalte W1 errechnet.
TEMP = NORM_SIGMOID(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden aus den Spalten W1, W2 und W3 errechnet.
TEMP = NORM_SIGMOID(W1[10:50]:W3)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils in Zeile 1-41 Werte enthalten. Die Werte von Spalte TEMP sind die normalisierten Werte von Zeile 10-50 von Spalte W1, die Werte von Spalte WX sind die normalisierten Werte von Zeile 10-50 von Spalte W2 und die Werte von Spalte WY sind die normalisierten Werte von Zeile 10-50 von Spalte W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden aus Zeile 10-50 von Spalte W1-W3 errechnet.
TEMP = NORM_SIGMOID(W1:W3, V4)
Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden aus der Spalte W4 errechnet.
TEMP = NORM_SIGMOID(W1:W3, W4:W8)
Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden aus den Spalten W4-W8 errechnet.

TEMP = NORM_SIGMOID(W1:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden für jede Spalte unabhängig errechnet (d. h. jeweils für Spalte W1, Spalte W2 usw. getrennt).
TEMP = NORM_SIGMOID(W1[10:50]:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils in Zeile 1-41 Werte enthalten. Die Werte von Spalte TEMP sind die normalisierten Werte von Zeile 10-50 von Spalte W1, die Werte von Spalte WX sind die normalisierten Werte von Zeile 10-50 von Spalte W2 und die Werte von Spalte WY sind die normalisierten Werte von Zeile 10-50 von Spalte W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden aus Zeile 10-50 von Spalte W1-W3 errechnet. Der Durchschnitt und die Standardabweichung für die Normalisierung werden für jede Spalte unabhängig errechnet.
TEMP = NORM_SIGMOID(W1:W3, W4:W6, COL)
Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden für jede Spalte unabhängig aus den Spalten W4-W6 errechnet (d. h. aus Spalte W4 für die Normalisierung von Spalte W1, aus Spalte W5 für die Normalisierung von Spalte W2 usw.).
TEMP = NORM_SIGMOID(W1:W3, ROW)
Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden unabhängig aus jeder Zeile in den Spalten W1, W2 und W3 errechnet.
TEMP = NORM_SIGMOID(W1[10:50]:W3, ROW)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils in Zeile 1-41 Werte enthalten. Die Werte von Spalte TEMP sind die normalisierten Werte von Zeile 10-50 von Spalte W1, die Werte von Spalte WX sind die normalisierten Werte von Zeile 10-50 von Spalte W2 und die Werte von Spalte WY sind die normalisierten Werte von Zeile 10-50 von Spalte W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden aus Zeile 10-50 von Spalte W1-W3 errechnet. Der Durchschnitt und die Standardabweichung für die Normalisierung werden aus jeder Zeile von Spalte W1-W3 errechnet.
TEMP = NORM_SIGMOID(W1:W3, W4:W10, ROW)
Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden unabhängig aus jeder Zeile in den Spalten W4-W10 errechnet.

Zugehörige Funktionen

Funktion	Beschreibung
NORM_MINMAX	Berechnet die Minimum/Maximum-Normalisierung eines Datenbereichs.
NORM_ZSCORE	Berechnet die z-Faktorwert-Normalisierung eines Datenbereichs.

NORM_ZSCORE



Syntax

NORM_ZSCORE(Daten [, Schlüsselwort]); NORM_ZSCORE(Daten, Durchschnitt, Stdabw [, Schlüsselwort]); NORM_ZSCORE(Daten, Basisdaten [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, die normalisiert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Durchschnitt, Stdabw

Diese beiden Parameter geben den Durchschnitt und die Standardabweichung für die Normalisierung an. Es muss sich um Konstanten handeln, außer bei dem Schlüsselwort ROW, wo es sich um Konstanten oder Spalten handeln kann.

Basisdaten

Dieser Parameter gibt einen Datenbereich an, der für die Berechnung des Durchschnitts und der Standardabweichung für die Normalisierung von Daten verwendet wird.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabebereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

NORM_ZSCORE berechnet die normalisierten Werte des angegebenen Datenbereichs. Die z-Faktorwert-Normalisierung wird wie folgt ausgeführt:

$$VX[y] = \frac{VX[y] - \text{mean}}{\text{std}}$$

Dabei werden *Durchschnitt* und *Stdabw* wie folgt festgelegt:

- Wenn Durchschnitt und Stdabw angegeben sind, werden diese Werte für den Durchschnitt bzw. die Standardabweichung verwendet. Wenn diese Parameter bei dem Schlüsselwort ROW angegeben sind, kann es sich bei Durchschnitt und Stdabw um Spalten handeln, die für jede Zeile von Daten einen Durchschnitt und eine Standardabweichung angeben. Wenn es sich bei Durchschnitt und Stdabw um Spalten handelt, müssen die Spalten entweder dieselbe Länge wie Daten aufweisen oder skalar sein (d. h., sie müssen einen Einzelwert enthalten, der als Konstante auf alle Werte in der entsprechenden Spalte von Daten angewendet wird).
- Wenn Basisdaten angegeben ist, werden der Durchschnitt und die Standardabweichung dieses Datenbereichs berechnet und für die Normalisierung von Daten verwendet. Die Spalten in Basisdaten müssen mindestens zwei Zellenwerte enthalten.
- Wenn keine dieser sich gegenseitig ausschließenden Optionen angegeben ist, werden der Durchschnitt und die Standardabweichung automatisch aus Daten errechnet.

NORM_ZSCORE gibt immer einen Datenbereich zurück, dessen Dimensionen denen des Eingabedatenbereichs entsprechen. Durchschnitt und Standardabweichung werden für jede Eingabespalte berechnet und für die Normalisierung der betreffenden Spalte verwendet.

Anmerkung: Wenn die Standardabweichung null ist, werden nur Nullen zurückgegeben.

Beispiele

TEMP = NORM_ZSCORE(COLUMN(3, 4, 5))
Erstellt eine neue Spalte TEMP, die die Werte -1,22, 0 und 1,22 enthält. (Durchschnitt und Standardabweichung [4 und 0,816] werden automatisch aus dem Datenbereich errechnet.)
TEMP = NORM_ZSCORE(COLUMN(3, 4, 5), 3.5, 1.2)
Erstellt eine neue Spalte TEMP, die die Werte -0,42, 0,42 und 1,25 enthält. (Hier sind Durchschnitt und Standardabweichung [3,5 und 1,2] als Argumente angegeben.)
TEMP = NORM_ZSCORE(W1)
Erstellt eine neue Spalte TEMP, die die normalisierten Werte des Inhalts von Spalte W1 enthält. Der Durchschnitt und die Standardabweichung für die Normalisierung werden aus der Spalte W1 errechnet.
TEMP = NORM_ZSCORE(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden für jede Spalte unabhängig errechnet (d. h. jeweils für Spalte W1, Spalte W2 usw. getrennt).
TEMP = NORM_ZSCORE(W1[10:50]:W3)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils in Zeile 1-41 Werte enthalten. Die Werte von Spalte TEMP sind die normalisierten Werte von Zeile 10-50 von Spalte W1, die Werte von Spalte WX sind die normalisierten Werte von Zeile 10-50 von Spalte W2 und die Werte von Spalte WY sind die normalisierten Werte von Zeile 10-50 von Spalte W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden aus Zeile 10-50 von Spalte W1-W3 errechnet. Der Durchschnitt und die Standardabweichung für die Normalisierung werden für jede Spalte unabhängig errechnet.

<p>TEMP = NORM_ZSCORE(W1:W3, W4:W6)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden für jede Spalte unabhängig aus den Spalten W4-W6 errechnet (d. h. aus Spalte W4 für die Normalisierung von Spalte W1, aus Spalte W5 für die Normalisierung von Spalte W2 usw.).</p>
<p>TEMP = NORM_ZSCORE(W1:W3, COL)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden für jede Spalte unabhängig errechnet (d. h. jeweils für Spalte W1, Spalte W2 usw. getrennt).</p>
<p>TEMP = NORM_ZSCORE(W1[10:50]:W3, COL)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY, die jeweils in Zeile 1-41 Werte enthalten. Die Werte von Spalte TEMP sind die normalisierten Werte von Zeile 10-50 von Spalte W1, die Werte von Spalte WX sind die normalisierten Werte von Zeile 10-50 von Spalte W2 und die Werte von Spalte WY sind die normalisierten Werte von Zeile 10-50 von Spalte W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden aus Zeile 10-50 von Spalte W1-W3 errechnet. Der Durchschnitt und die Standardabweichung für die Normalisierung werden für jede Spalte unabhängig errechnet.</p>
<p>TEMP = NORM_ZSCORE(W1[10:50]:W3, COL)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden für jede Spalte unabhängig aus den Spalten W4-W6 errechnet (d. h. aus Spalte W4 für die Normalisierung von Spalte W1, aus Spalte W5 für die Normalisierung von Spalte W2 usw.).</p>
<p>TEMP = NORM_ZSCORE(W1:W3, ROW)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden unabhängig aus jeder Zeile in den Spalten W1, W2 und W3 errechnet.</p>
<p>TEMP = NORM_ZSCORE(W1[10:50]:W3, ROW)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY, die jeweils in Zeile 1-41 Werte enthalten. Die Werte von Spalte TEMP sind die normalisierten Werte von Zeile 10-50 von Spalte W1, die Werte von Spalte WX sind die normalisierten Werte von Zeile 10-50 von Spalte W2 und die Werte von Spalte WY sind die normalisierten Werte von Zeile 10-50 von Spalte W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden aus Zeile 10-50 von Spalte W1-W3 errechnet. Der Durchschnitt und die Standardabweichung für die Normalisierung werden aus jeder Zeile von Spalte W1-W3 errechnet.</p>
<p>TEMP = NORM_ZSCORE(W1:W3, W4:W10, ROW)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY. Diese Spalten enthalten die normalisierten Werte des Inhalts von Spalte W1, W2 bzw. W3. Der Durchschnitt und die Standardabweichung für die Normalisierung werden unabhängig aus jeder Zeile in den Spalten W4-W10 errechnet.</p>

Zugehörige Funktionen

Funktion	Beschreibung
NORM_MINMAX	Berechnet die Minimum/Maximum-Normalisierung eines Datenbereichs.
NORM_SIGMOID	Berechnet die sigmoidale Normalisierung eines Datenbereichs.

NOT

Syntax

NOT(Daten); ! Daten

Parameter

Daten

Die numerischen Werte, deren logisches Nicht berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

NOT gibt das logische Nicht der Werte im angegebenen Datenbereich zurück. NOT gibt für jede Eingabespalte eine neue Spalte zurück, die das logische Nicht der Werte in der entsprechenden Eingabespalte enthält. Diese Funktion gibt bei Werten ungleich null eine Null und bei Werten gleich null eine Eins zurück.

Anmerkung: Der Operator NOT kann durch ein Ausrufezeichen (!) abgekürzt werden. Stellen Sie das Ausrufezeichen dem Datenwert voran (statt z. B. NOT(W1) anzugeben, können Sie einfach !W1 eingeben).

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = NOT(3,2) oder TEMP = !1
Erstellt eine neue Spalte TEMP, die den Wert null enthält.
TEMP = !0 oder TEMP = !(2+2=3)
Erstellt eine neue Spalte TEMP, die den Wert eins enthält.
TEMP = !W1
Erstellt eine neue Spalte TEMP, in der jeder Wert das logische Nicht der Werte in Spalte W1 darstellt.
TEMP = !W1:W3
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils das logische Nicht der Werte in Spalte W1, die Werte von Spalte WX sind das logische Nicht der Werte in Spalte W2 und die Werte von Spalte WY sind das logische Nicht der Werte in Spalte W3.
TEMP = !W1[10:20]
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils das logische Nicht der Werte in Zeile 10-20 von Spalte W1 darstellen. Die anderen Zellen in TEMP sind leer.

TEMP = !W1[1:5]:W2

Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind das logische Nicht der Werte der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind das logische Nicht der Werte der entsprechenden Zeilen von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
AND	Berechnet das logische Und zwischen den beiden angegebenen Datenbereichen.
INVERSE	Berechnet den Negativwert des Inhalts des angegebenen Datenbereichs.
OR	Berechnet das logische Oder zwischen den beiden angegebenen Datenbereichen.
SIGN	Berechnet das Vorzeichen (positiv oder negativ) der Werte im angegebenen Datenbereich.

NPV



Syntax

NPV(Daten, Satz [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, die den erwarteten Nettokapitalfluss darstellen, aus dem der aktuelle Nettowert errechnet wird. Dabei kann es sich um eine Zeile, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt.

Satz

Der numerische Wert, der den Rabattsatz für die Länge eines Zeitraums darstellt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabedatenbereich angewendet wird. Ist kein Schlüsselwort angegeben, wird ROW als Standard verwendet. Wählen Sie eine der folgenden Angaben aus:

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Beschreibung

NPV berechnet den aktuellen Nettowert einer Investition aufgrund einer Folge periodischer Kapitalflüsse und eines Rabattsatzes. Der aktuelle Nettowert einer Investition ist der aktuelle Wert einer Folge zukünftiger Zahlungen (negative Werte) und Erträge (positive Werte). NPV wird nach der folgenden Formel berechnet:

$$\text{NPV} = \sum_{t=1}^n \frac{\text{data}_t}{(1 + \text{rate})^t}$$

Dabei ist n die Anzahl der Kapitalflüsse (Datenwerte).

Die Reihenfolge der Datenwerte dient dazu, die Reihenfolge der Kapitalflüsse zu interpretieren. Die NPV-Investition beginnt einen Zeitraum vor dem Datum des ersten Kapitalflusswerts und endet bei dem letzten Kapitalfluss in der Liste.

Anmerkung: Wenn der erste Kapitalfluss zu Beginn des ersten Zeitraums auftritt, muss der erste Wert zum NPV-Ergebnis addiert werden, da er nicht in den Datenwerten enthalten ist.

Die Anzahl der von NPV zurückgegebenen Spalten hängt von Schlüsselwort ab.

- Wenn Schlüsselwort den Wert COL hat, gibt NPV für jede Eingabespalte eine neue Spalte zurück. Jede neue Spalte enthält einen Wert (den aktuellen Nettowert aller Zellen in der entsprechenden Eingabespalte).
- Wenn Schlüsselwort den Wert ROW hat, gibt NPV eine neue Spalte zurück, die den aktuellen Nettowert für jede Zeile von Daten enthält.

Anmerkung: Leere Zellen werden bei der Berechnung des aktuellen Nettowerts ignoriert.

Beispiele

<code>TEMP = NPV(W1:W3, .10)</code> oder <code>TEMP = NPV(W1:W3, .10, ROW)</code>
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der den aktuellen Nettowert der Werte der Spalten W1, W2 und W3 darstellt. Der verwendete Rabattsatz beträgt 10 %.
<code>TEMP = NPV(W1, .10, COL)</code>
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der den aktuellen Nettowert des Inhalts von Spalte W1 bei einem Rabattsatz von 10 % darstellt.
<code>TEMP = NPV(W1:W3, .10) - 1000</code>
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der den aktuellen Nettowert des Inhalts der Spalten W1, W2 und W3 bei einer Anfangszahlung von 1000 darstellt. Der Rabattsatz beträgt 10 %.
<code>TEMP = NPV(W1[10:20], .10, COL)</code>
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der den aktuellen Nettowert der Zellen in Zeile 10-20 von Spalte W1 darstellt. Der Rabattsatz beträgt 10 %.

```
TEMP = NPV(W1[1:5]:W4, .10)
```

Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der den aktuellen Nettowert der Zellen in Zeile 1-5 von Spalte W1 bis W4 darstellt.

NUMBER

Syntax

NUMBER(Daten [, Schlüsselwort])

Parameter

Daten

Die ASCII-Textdaten, die in numerische Werte konvertiert werden sollen. Dabei kann es sich um ASCII-Text in Anführungszeichen, eine Textspalte, einen Zellenbereich mit Text oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie Textformate für Datums- und Uhrzeitangaben interpretiert werden sollen. Wählen Sie eines der Schlüsselwörter in der folgenden Tabelle aus.

Anmerkung: Wird dieser Parameter nicht angegeben, ist der Standard 1.

Schlüsselwort	Format	Beschreibung
0	#####	Konvertiert die ersten 5 Zeichen jeder Zeichenfolge in eine eindeutige Zahl.
1	€ (Standard)	Konvertiert Eurowerte in numerische Werte (z. B. "123,45 €" in 123,45).
2	%	Konvertiert Prozentwerte in numerische Werte (z. B. "50 %" in 0,5).
3	mm/tt/jj hh:mm	Konvertiert ein Datum und eine Uhrzeit in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (zum Jahr jj wird automatisch 1900 addiert).
4	tt-mmm-jj	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (zum Jahr jj wird automatisch 1900 addiert).

5	mm/tt/jj	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (zum Jahr jj wird automatisch 1900 addiert).
6	mmm-jj	Konvertiert ein Datum in die Anzahl der Tage, die zwischen dem ersten Tag des angegebenen Monats und dem 1. Januar 0000 vergangen sind (zum Jahr jj wird automatisch 1900 addiert).
7	tt-mmm	Konvertiert ein Datum in die Anzahl der Tage, die seit dem Anfang des Jahres vergangen sind (z. B. "01-FEB" in 32).
8	mmm	Konvertiert eine Monatsabkürzung aus 3 Buchstaben in einen Wert zwischen 1 und 12 (z. B. "NOV" in 11).
9	{Januar Februar März ... }	Konvertiert einen vollständigen Monatsnamen in einen Wert zwischen 1 und 12 (z. B. "November" in 11).
10	{Son Mon Die ... }	Konvertiert eine Wochentagsabkürzung aus 3 Buchstaben in einen Wert zwischen 0 und 6, wobei die Woche am Sonntag beginnt (z. B. "Son" in 0).
11	{Sonntag Montag Dienstag ... }	Konvertiert einen vollständigen Wochentagsnamen in einen Wert zwischen 0 und 6, wobei die Woche am Sonntag beginnt (z. B. "Montag" in 1).
12	hh:mm:ss {AM PM}	Konvertiert die Uhrzeit in die Anzahl der Sekunden, die seit Mitternacht vergangen sind (z. B. "01:00:00 AM" in 3600).
13	hh:mm:ss	Konvertiert die Uhrzeit in die Anzahl der Sekunden, die seit Mitternacht vergangen sind (z. B. "01:00:00" in 3600).
14	hh:mm {AM PM}	Konvertiert die Uhrzeit in die Anzahl der Minuten, die seit Mitternacht vergangen sind (z. B. "01:00 AM" in 60).
15	hh:mm	Konvertiert die Uhrzeit in die Anzahl der Minuten, die seit Mitternacht vergangen sind (z. B. "01:00" in 60).

16	mm:ss	Konvertiert die Uhrzeit in die Anzahl der Sekunden, die seit Mitternacht vergangen sind (z. B. "30:00" in 1800).
17	ttmm	Konvertiert ein Datum in die Anzahl der Tage, die seit dem Anfang des Jahres vergangen sind (z. B. "3101" in 31).
18	ttmmm	Konvertiert ein Datum in die Anzahl der Tage, die seit dem Anfang des Jahres vergangen sind (z. B. "31JAN" in 31).
19	ttmmmjj	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (wenn das Jahr jj kleiner-gleich 20 ist, wird automatisch 1900 addiert; andernfalls wird 2000 addiert).
20	ttmmmjjjj	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (z. B. "31JAN0000" in 31).
21	ttmmjj	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (wenn das Jahr jj kleiner-gleich 20 ist, wird automatisch 1900 addiert; andernfalls wird 2000 addiert).
22	ttmmjjjj	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (z. B. "31010000" in 31).
23	mmtt	Konvertiert ein Datum in die Anzahl der Tage, die seit dem Anfang des Jahres vergangen sind (z. B. "0131" in 31).
24	mmttjj	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (wenn das Jahr jj kleiner-gleich 20 ist, wird automatisch 1900 addiert; andernfalls wird 2000 addiert).

25	mmttjjjj	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (z. B. "01010001" in 366).
26	mmm	Konvertiert eine Monatsabkürzung aus 3 Buchstaben in einen Wert zwischen 1 und 12 (z. B. "NOV" in 11) [identisch mit Schlüsselwort 8].
27	mmmtt	Konvertiert ein Datum in die Anzahl der Tage, die seit dem Anfang des Jahres vergangen sind (z. B. "JAN31" in 31).
28	mmmttjj	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (wenn das Jahr jj kleiner-gleich 20 ist, wird automatisch 1900 addiert; andernfalls wird 2000 addiert).
29	mmmttjjjj	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (z. B. "FEB010000" in 32).
30	mmmj	Konvertiert ein Datum in die Anzahl der Tage, die zwischen dem ersten Tag des angegebenen Monats und dem 1. Januar 0000 vergangen sind (wenn das Jahr jj kleiner-gleich 20 ist, wird automatisch 1900 addiert; andernfalls wird 2000 addiert).
31	mmmjjjj	Konvertiert ein Datum in die Anzahl der Tage, die zwischen dem ersten Tag des angegebenen Monats und dem 1. Januar 0000 vergangen sind (z. B. "FEB010000" in 32).
32	mmjj	Konvertiert ein Datum in die Anzahl der Tage, die zwischen dem ersten Tag des angegebenen Monats und dem 1. Januar 0000 vergangen sind (wenn das Jahr jj kleiner-gleich 20 ist, wird automatisch 1900 addiert; andernfalls wird 2000 addiert).

33	mmjjjj	Konvertiert ein Datum in die Anzahl der Tage, die zwischen dem ersten Tag des angegebenen Monats und dem 1. Januar 0000 vergangen sind (z. B. "020000" in 32).
34	jjmm	Konvertiert ein Datum in die Anzahl der Tage, die zwischen dem ersten Tag des angegebenen Monats und dem 1. Januar 0000 vergangen sind (wenn das Jahr jj kleiner-gleich 20 ist, wird automatisch 1900 addiert; andernfalls wird 2000 addiert).
35	jjmmtt	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (wenn das Jahr jj kleiner-gleich 20 ist, wird automatisch 1900 addiert; andernfalls wird 2000 addiert).
36	jjmmm	Konvertiert ein Datum in die Anzahl der Tage, die zwischen dem ersten Tag des angegebenen Monats und dem 1. Januar 0000 vergangen sind (wenn das Jahr jj kleiner-gleich 20 ist, wird automatisch 1900 addiert; andernfalls wird 2000 addiert).
37	jjmmtt	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (wenn das Jahr jj kleiner-gleich 20 ist, wird automatisch 1900 addiert; andernfalls wird 2000 addiert).
38	jjjj	Konvertiert das Jahr in die Anzahl der Jahre, die seit dem Jahr 0000 vergangen sind (z. B. "1998" in 1998).
39	jjjjmm	Konvertiert ein Datum in die Anzahl der Tage, die zwischen dem ersten Tag des angegebenen Monats und dem 1. Januar 0000 vergangen sind (z. B. "000102" in 32).

40	jjjjmmtt	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (z. B. "00000201" in 32).
41	jjjjmmm	Konvertiert ein Datum in die Anzahl der Tage, die zwischen dem ersten Tag des angegebenen Monats und dem 1. Januar 0000 vergangen sind (z. B. "000102" in 32).
42	jjjjmmtt	Konvertiert ein Datum in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (z. B. "0000FEB01" in 32).
43	<Tag>* <Monat>	Konvertiert ein Datum mit Begrenzer, das aus Tag und Monat besteht, in die Anzahl der Tage, die seit dem Anfang des Jahres vergangen sind (z. B. "15-JAN" in 15).
44	<Tag>* <Monat>* <Jahr>	Konvertiert ein Datum mit Begrenzer, das aus Tag, Monat und Jahr besteht, in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (z. B. "1/1/0001" in 366).
45	<Monat>* <Tag>	Konvertiert ein Datum mit Begrenzer, das aus Monat und Tag besteht, in die Anzahl der Tage, die seit dem Anfang des Jahres vergangen sind (z. B. "JAN 31" in 31).
46	<Monat>* <Tag>* <Jahr>	Konvertiert ein Datum mit Begrenzer, das aus Monat, Tag und Jahr besteht, in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (z. B. "JAN 1, 0001" in 366).
47	<Monat>* <Jahr>	Konvertiert ein Datum mit Begrenzer, das aus Monat und Jahr besteht, in die Anzahl der Tage, die zwischen dem ersten Tag des angegebenen Monats und dem 1. Januar 0000 vergangen sind.
48	<Jahr>* <Monat>	Konvertiert ein Datum mit Begrenzer, das aus Jahr und Monat besteht, in die Anzahl der Tage, die zwischen dem ersten Tag des angegebenen Monats und dem 1. Januar 0000 vergangen sind.

49	<Jahr>* <Monat>* <Tag>	Konvertiert ein Datum mit Begrenzer, das aus Jahr, Monat und Tag besteht, in die Anzahl der Tage, die seit dem 1. Januar 0000 vergangen sind (z. B. "0001/01/01" in 366).
50	jj	Konvertiert das Jahr in die Anzahl der Jahre, die seit dem Jahr 0000 vergangen sind (z. B. "97" in 97).
51	mm	Konvertiert den Monat in einen Wert zwischen 1 und 12 (z. B. "SEP" in 9).
52	tt	Konvertiert den Tag in einen Wert zwischen 1 und 31 (z. B. "28" in 28).
53	{Januar Februar März ... }	Konvertiert einen vollständigen Monatsnamen in einen Wert zwischen 1 und 12 (z. B. "November" in 11) [identisch mit Schlüsselwort 9].
54	{Sonntag Montag Dienstag ... }	Konvertiert einen vollständigen Wochentagsnamen in einen Wert zwischen 1 und 7, wobei die Woche am Sonntag beginnt (z. B. "Sonntag" in 1).
55	{Son Mon Die ... }	Konvertiert eine Wochentagsabkürzung aus 3 Buchstaben in einen Wert zwischen 1 und 7, wobei die Woche am Sonntag beginnt (z. B. "Son" in 1).

Beschreibung

NUMBER konvertiert Textwerte im angegebenen Datenbereich anhand des angegebenen Umwandlungsformats für Datums- und Uhrzeitangaben in numerische Werte. Wenn eine Zeichenfolge mit dem angegebenen Schlüsselwort nicht geparkt werden kann, gibt NUMBER einen Fehler zurück. Format 0 konvertiert die ersten fünf Zeichen jeder Zeichenfolge für jede eindeutige Zeichenfolge in eine unterschiedliche Zahl. Dies ist ein einfacher Weg, eine Textspalte in eindeutige Klassen umzuwandeln, die als Ausgaben für einen Klassifikator dienen können.

Die begrenzten Formate (Schlüsselwort 43-49) unterstützen die folgenden Begrenzer:

- / (Schrägstrich)
- - (Strich)
- , (Komma)
- " " (Leerzeichen)
- : (Doppelpunkt)

Monate können als mm oder mmm dargestellt werden; Tage können als t oder tt dargestellt werden; Jahre können als jj oder jjjj dargestellt werden.

Anmerkung: Alle Jahre in Datumsangaben können als jjjj statt als jj angegeben werden, um das Jahr 2000 zu unterstützen. Im Interesse der Abwärtskompatibilität wird bei Schlüsselwort 1-16 zu 2-stelligen Jahresangaben jj automatisch 1900 addiert. Bei Schlüsselwort 17-55 wird zu jj < Grenzwert automatisch 2000 addiert; zu jj ≥ Grenzwert wird automatisch 1900 addiert.

Anmerkung: Der Jahr-2000-Grenzwert wird auf der Registerkarte **Datenbereinigung** des Fensters **Erweiterte Einstellungen** festgelegt (Aufruf über **Optionen > Einstellungen > Erweiterte Einstellungen**).

Anmerkung: Wenn Sie den Jahr-2000-Schwellenwert ändern, müssen Sie alle Makrofunktionen aktualisieren, die mithilfe der Makrofunktion NUMBER Datumswerte mit 2-stelligen Jahren bearbeiten. Um eine Aktualisierung einer Makrofunktion zu erzwingen, können Sie eine beliebige Bearbeitung vornehmen (z. B. ein Leerzeichen hinzufügen und wieder löschen) und auf das Hakensymbol klicken, damit die Änderung übernommen wird.

Anmerkung: Bei Format 0 werden nur die ersten fünf Zeichen jeder Zeichenfolge für die Generierung einer eindeutigen Zahl herangezogen. Alle Zeichenfolgen mit denselben ersten fünf Zeichen ergeben denselben numerischen Wert. Dieselbe Zeichenfolge ergibt jedes Mal denselben numerischen Wert, sogar in verschiedenen Arbeitsblättern. Bearbeiten Sie die Zeichenfolgen bei Bedarf mit Zeichenfolgemakros, sodass die ersten fünf Zeichen eine Klasse eindeutig definieren. Beachten Sie, dass die resultierenden numerischen Werte unter Umständen sehr klein sind. Mithilfe des Fensters **Anzeigeformate** können Sie entweder die Anzahl der angezeigten Dezimalstellen erhöhen oder das Format in den exponentiellen Modus (00E+00) ändern.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

<code>TEMP = NUMBER("1,23 €")</code> oder <code>TEMP = NUMBER("123 %", 2)</code> Erstellt eine neue Spalte TEMP, die den Wert 1,23 enthält.
<code>TEMP = NUMBER(column("Jan", "Apr", "Nov", 8))</code> Erstellt eine neue Spalte TEMP, die die Werte 1, 4 und 11 enthält.
<code>TEMP = NUMBER("1:52 PM", 14)</code> Erstellt eine neue Spalte TEMP, die die Zahl 832 enthält.
<code>TEMP = NUMBER("1/1/95", 5)</code> Erstellt eine neue Spalte TEMP, die die Zahl 728660 enthält.
<code>TEMP = NUMBER(W1)</code> Erstellt eine neue Spalte TEMP, die die numerischen Werte der Zeichenfolgen in Spalte W1 enthält. Eurowerte werden korrekt in numerische Werte konvertiert. Bei Zeichenfolgen, die mit dem \$-Format nicht geparkt werden können, wird ??? zurückgegeben.

<p>TEMP = NUMBER(W1:W3, 4)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY. Die Spalte TEMP enthält die numerischen Werte der Zeichenfolgen in Spalte W1. Die Spalte WX enthält die numerischen Werte der Zeichenfolgen in Spalte W2. Die Spalte WY enthält die numerischen Werte der Zeichenfolgen in Spalte W3. Alle Datumsangaben in dem Format tt-mmm-jj werden in die Anzahl der Tage seit dem 1. Januar 0000 konvertiert. Bei Zeichenfolgen, die mit dem \$-Format nicht geparkt werden können, wird ??? zurückgegeben.</p>
<p>TEMP = NUMBER(W1[10:20]:W2, 10)</p> <p>Erstellt zwei neue Spalten TEMP und WX. Die Spalte TEMP enthält die numerischen Werte der Zeichenfolgen in Zeile 10-20 von Spalte W1. Die Spalte WX enthält die numerischen Werte der Zeichenfolgen in Zeile 10-20 von Spalte W2. Alle standardmäßigen Darstellungen der Wochentage durch drei Zeichen werden in die Zahlen 0 bis 6 konvertiert (0 = Sonntag, 6 = Samstag). Wenn es für einen Wochentag keine Übereinstimmung gibt, wird ??? zurückgegeben.</p>
<p>TEMP = NUMBER(W1, 0)</p> <p>Sofern Spalte W1 nur 5-stellige Zeichenfolgen enthält, wird eine neue Spalte TEMP erstellt, die für jede eindeutige Zeichenfolge einen unterschiedlichen numerischen Wert enthält.</p>

Zugehörige Funktionen

Funktion	Beschreibung
WEEKDAY	Konvertiert ASCII-Datumszeichenfolgen in Wochentage.

OFFSET



Syntax

OFFSET(Daten)

Parameter

Daten

Die Werte, deren Offset berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

OFFSET gibt den Offset der Werte des angegebenen Datenbereichs vom ersten Wert zurück. OFFSET gibt eine neue Spalte mit den Offsetwerten zurück, die bei eins beginnen und sich bis zur Länge der längsten Spalte im Datenbereich fortsetzen.

Beispiele

TEMP = OFFSET(5)
Erstellt eine neue Spalte TEMP, die den Wert eins enthält.
TEMP = OFFSET(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert den Offset der Spalte W1 darstellt (von eins bis zur Länge der Spalte W1).
TEMP = OFFSET(W1:W3)
Erstellt eine neue Spalte TEMP, in der jeder Wert den Offset darstellt (von eins bis zur längsten Spalte von W1, W2 oder W3).

Zugehörige Funktionen

Funktion	Beschreibung
COUNT	Zählt die Anzahl der Zellen, die Werte enthalten, im angegebenen Datenbereich.
DELAY	Gibt die Verzögerung der Eingabespaltenwerte um eine angegebene Anzahl von Zeitschritten an.

OR

Syntax

Daten1 OR Daten2; Daten1 || Daten2

Parameter

Daten1

Die Zahlen, die durch logisches Oder mit den Werten in Daten2 verknüpft werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makro-funktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Die Zahl(en), die durch logisches Oder mit den Werten in Daten1 verknüpft werden soll(en). Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Daten2 muss mit der Anzahl der Spalten in Daten1 übereinstimmen, es sei denn, bei Daten2 handelt es sich um eine Konstante. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makro-funktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

OR berechnet das logische Oder zwischen den beiden angegebenen Datenbereichen. OR gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten1 durch logisches Oder mit der entsprechenden Spalte von

Daten2 verknüpft wird (d. h., die erste Spalte von Daten1 wird durch logisches Oder mit der ersten Spalte von Daten2 verknüpft, die zweite Spalte mit der zweiten Spalte usw.).

Wenn es sich bei Daten2 um eine Konstante handelt, wird jeder Wert in Daten1 durch logisches Oder mit dem betreffenden Wert verknüpft. Wenn Daten2 eine oder mehrere Spalten enthält, werden die Berechnungen zeilenweise zwischen einer Spalte von Daten1 und einer Spalte von Daten2 ausgeführt. Die erste Zeile von Daten1 wird durch logisches Oder mit der ersten Zeile von Daten2 verknüpft, die zweite Zeile mit der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Wenn eine Spalte in jeder Zeile dieselbe Zahl x wie Daten2 enthält, ist dies dasselbe, als wenn als Daten2 die Konstante x verwendet wird.

Anmerkung: Der Operator OR kann durch einen doppelten vertikalen Strich (||) abgekürzt werden. Mit dem doppelten vertikalen Strich können die beiden Argumente getrennt werden (statt z. B. W1 OR 3 anzugeben, können Sie einfach W1 || 3 eingeben).

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = 1 OR 8 oder TEMP = 1 8
Erstellt eine neue Spalte TEMP, die den Wert eins enthält (jede Zahl ungleich null wird als eins behandelt).
TEMP = W1 1
Erstellt eine neue Spalte TEMP, die nur Einsen enthält (jeder Wert, der durch logisches Oder mit der Zahl eins verknüpft wird, ergibt eins).
TEMP = W1 W2
Erstellt eine neue Spalte TEMP, in der jeder Zeilenwert von Spalte W1 durch logisches Oder mit dem entsprechenden Zeilenwert von Spalte W2 verknüpft ist.
TEMP = W1:W3 W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Werte von Spalte W1 durch logisches Oder mit den entsprechenden Zeilenwerten von Spalte W4 verknüpft. In der Spalte WX sind die Werte von Spalte W2 durch logisches Oder mit den Werten von Spalte W5 verknüpft. In der Spalte WY sind die Werte von Spalte W3 durch logisches Oder mit den Werten von Spalte W6 verknüpft.
TEMP = W1[10:20] W2
Erstellt eine neue Spalte TEMP, in deren ersten 11 Zellen die Werte in Zeile 10-20 der Spalten W1 und W2 durch logisches Oder verknüpft sind. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
AND	Berechnet das logische Und zwischen den beiden angegebenen Datenbereichen.
NOT	Berechnet das logische Nicht des Inhalts des angegebenen Datenbereichs.



Syntax

PCA(Daten)

Parameter

Daten

Die numerischen Werte, deren Hauptkomponenten berechnet werden sollen. Dabei kann es sich um eine Konstante, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt.

Beschreibung

PCA wendet auf den angegebenen Datenbereich eine Analyse der Hauptkomponente (Principal Component Analysis, PCA) an. PCA ermittelt die orthogonalen Eigenvektoren zu dem durch Daten angegebenen Datenbereich mithilfe einer Einzelwertdekomposition. PCA gibt für jede der n Eingabespalten eine neue Spalte zurück, außerdem eine zusätzliche Spalte. Die ersten n Spalten enthalten die Eigenvektoren (jeder Eigenvektor wird als eine Zeile in den n Spalten gelesen). Die letzte Ausgabespalte enthält die entsprechenden Magnituden der Eigenwerte. Die Eigenvektoren werden gemäß den entsprechenden Eigenwerten angeordnet.

Anmerkung: Fehlende Werte (z. B. leere Zellen oder ???-Zellen) werden als Nullen gewertet. Kürzere Spalten in Daten werden mit Nullen bis zur Länge der längsten Spalte aufgefüllt.

Die PCA wird wie folgt berechnet:

- Jede der k Zeilen von Daten ist ein n -dimensionaler Vektor v_i (n ist die Anzahl der Spalten in Daten). Diese werden wie folgt zur Berechnung der Korrelationsmatrix A herangezogen:

$$A = \sum_{i=1}^k v_i v_i^T$$

- Die n -mal- n -Korrelationsmatrix A wird mithilfe einer Einzelwertdekomposition in drei Matrices zerlegt:

$$A = U \Sigma U^T$$

Die Zeilen von U sind die Eigenvektoren von A und Σ ist eine diagonale Matrix, in der jedes diagonale Element die Magnitude der Eigenwerte für A darstellt.

Die Makrofunktion PCA gibt U in den ersten n Spalten und die diagonalen Element von Σ in der letzten Spalte zurück.

Beispiele

TEMP = PCA(5)	Erstellt zwei neue Spalten TEMP und WX, die den Wert -1 bzw. 0 enthalten.
TEMP = PCA(W1)	Erstellt zwei neue Spalten TEMP und WX. Die Spalte TEMP enthält den Wert eins, die Spalte WX enthält den entsprechenden Eigenwert.
TEMP = PCA(W1:W3)	Erstellt vier neue Spalten TEMP, WX, WY und WZ. Die Werte in den ersten drei Spalten enthalten für die Daten in Spalte W1-W3 einen Eigenvektor pro Zeile. Der Wert in Spalte WZ enthält die entsprechenden Eigenwerte.

Zugehörige Funktionen

Funktion	Beschreibung
PCA_FEATURES	Extrahiert n Komponenten aus dem angegebenen Datenbereich.

PCA_FEATURES



Syntax

PCA_FEATURES(Komponentenzahl, Daten [, PCA(Basisdaten)])

Parameter

Komponentenzahl

Die Anzahl der Komponenten, die mithilfe einer Analyse der Hauptkomponente (Principal Component Analysis, PCA) aus dem angegebenen Datenbereich extrahiert werden sollen. Dieser Wert muss eine positive Ganzzahl zwischen eins und der Anzahl der Spalten in dem durch Daten angegebenen Datenbereich sein.

Daten

Die numerischen Werte, aus denen Komponenten extrahiert werden sollen. Dabei kann es sich um eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

PCA(Basisdaten)

Wenn dieser optionale Parameter angegeben ist, wird PCA auf diesen Datenbereich Basisdaten angewendet; die resultierenden Eigenvektoren werden dazu verwendet, Komponenten aus dem Datenbereich Daten zu extrahieren. Informationen zur Formatdefinition von Basisdaten (identisch mit Daten) finden Sie im Abschnitt "Mak-

rofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt. Die Anzahl der Spalten in Basisdaten muss mit der Anzahl der Spalten in Daten übereinstimmen.

Beschreibung

PCA_FEATURES extrahiert die obersten Komponentenzahl Komponenten aus dem angegebenen Datenbereich. Dabei werden Komponentenzahl Spalten zurückgegeben, die die von der Analyse der Hauptkomponente des Datenbereichs Basisdaten (falls angegeben) generierten Eigenvektoren verwenden. Falls nicht angegeben, dient der Datenbereich Daten zur Generierung der Eigenvektoren. In diesem Fall wird Daten vor der Analyse der Hauptkomponente automatisch bei Mittelwert null und Einheitsvarianz normalisiert.

Die Komponenten werden wie folgt berechnet:

- Der Datenbereich Daten wird automatisch bei Mittelwert null und Einheitsvarianz normalisiert. Mit anderen Worten,
PCA_FEATURES(Komponentenzahl, Daten)

ist äquivalent zu

PCA_FEATURES(Komponentenzahl, Daten, PCA(Daten, COL))

Eine Normalisierung von Daten erfolgt nicht automatisch. Um Daten mithilfe von NORM_ZSCORE zu normalisieren, können Sie Folgendes angeben:

PCA_FEATURES(Komponentenzahl, Daten, PCA(NORM_ZSCORE(Daten, COL)))

- Auf den normalisierten Datenbereich wird eine Analyse der Hauptkomponente angewendet, um dessen Eigenvektoren zu generieren (nähere Details finden Sie in der Beschreibung der Makrofunktion PCA). Dies erfolgt automatisch für Daten, wenn Basisdaten nicht angegeben ist. Es erfolgt durch den expliziten Aufruf der Makrofunktion PCA, wenn Basisdaten angegeben ist.
- Jede Zeile (v_i) des Datenbereichs (Daten) wird in ein neues Koordinatensystem (u_i) transformiert, basierend auf den Komponentenzahl (m) höchstrangigen Eigenvektoren, die E_m bilden:

$$u_i = E_m v_i = \begin{bmatrix} E_{11} & \dots & E_{1n} \\ \dots & \dots & \dots \\ E_{m1} & \dots & E_{mn} \end{bmatrix} \begin{bmatrix} v_1 \\ \dots \\ v_n \end{bmatrix}$$

- Die k Zeilen der transformierten Daten (u_1 für u_k) werden zurückgegeben (n Spalten).

Wenn der Datenbereich Basisdaten angegeben ist, muss er dieselbe Anzahl von Spalten wie der Datenbereich Daten aufweisen; andernfalls wird ein Fehler zurückgegeben.

Anmerkung: Da die PCA-Berechnung für einen Datenbereich ziemlich rechenintensiv sein kann, ist es wesentlich effizienter, die Makrofunktion BUFFER für die PCA-Berechnung einzusetzen. Beispiel: PCA_FEATURES(Komponentenzahl, Bereich, BUFFER(PCA(Basisdaten)))

Beispiele

<p>TEMP = PCA_FEATURES(5, W1:W7) Erstellt fünf neue Spalten TEMP, WW, WX, WY und WZ, die die fünf Hauptkomponenten des Datenbereichs W1:W7 enthalten. Der Datenbereich W1:W7 fungiert als Basis für die Transformation.</p>
<p>TEMP = PCA_FEATURES(3, W1:W4, PCA(W10:W13))</p> <p>Erstellt drei neue Spalten TEMP, WX und WY, die die drei Hauptkomponenten des Datenbereichs W1:W4 enthalten. Der Datenbereich W10:W13 fungiert als Basis für die Transformation.</p>
<p>TEMP = PCA_FEATURES(3, W1:W4, BUFFER(PCA(W10:W13)))</p> <p>Erstellt drei neue Spalten TEMP, WX und WY, die die drei Hauptkomponenten des Datenbereichs W1:W4 enthalten. Der Datenbereich W10:W13 fungiert als Basis für die Transformation. Nachdem die Hauptkomponenten des Datenbereichs W10:W13 berechnet worden sind, werden die betreffenden Werte als Konstanten gespeichert. Wenn sich die Datenwerte in den Spalten W10 bis W13 ändern, wirkt sich dies nicht auf diese Funktionsdefinition aus.</p>

Zugehörige Funktionen

Funktion	Beschreibung
PCA	Berechnet die Eigenvektoren für Hauptkomponenten des angegebenen Datenbereichs.

POSITION

Syntax

POSITION(Spaltenname, Muster [, Start [, Vorkommen]])

Parameter

Spaltenname

Der Wert einer Spalte (muss dem Datentyp string angehören).

Muster

Das Muster (die Zeichenfolge), nach dem gesucht wird.

Start

Das Byte, bei dem die Suche beginnen soll.

Vorkommen

Wenn Sie einen Wert für n angeben, wird nach dem n-ten Vorkommen des Musters gesucht.

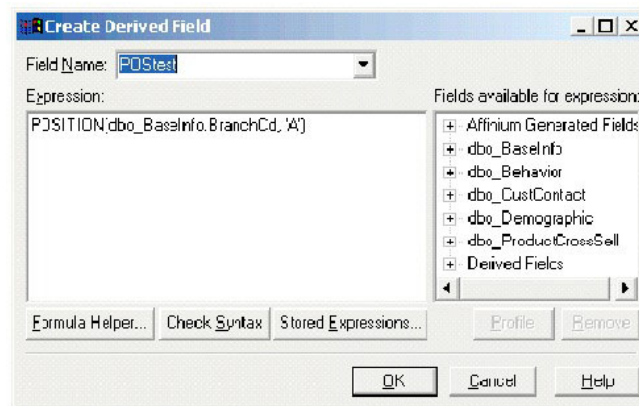
Beschreibung

POSITION gibt die Startbyteposition eines Musters oder einer Zeichenfolge innerhalb des Werts einer Spalte (Spaltenname), die dem Zeichenfolgedatentyp angehören muss, zurück. Wenn Start angegeben ist, beginnt die Suche dort. Vorkommen ist das n-te Vorkommen des Musters, das zurückgegeben werden soll.

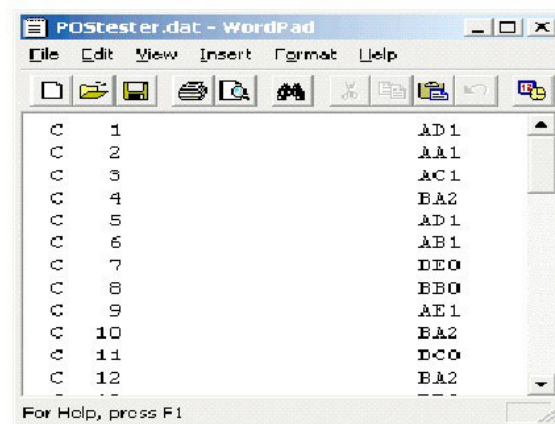
Anmerkung: Bei der Suche wird die Groß-/Kleinschreibung nicht beachtet.

Beispiele

Im folgenden Beispiel wird nach dem Muster oder der Zeichenfolge 'A' innerhalb des Werts der Spalte dbo_BaseInfo.BranchCd gesucht, der Rückgabewert wird dem abgeleiteten Feld POStest zugeordnet.



Das folgende Beispiel zeigt einige Zeilen aus der Tabelle, wobei die Werte von dbo_BaseInfo.BranchCd und POStest nebeneinander zu sehen sind.



Ein komplexeres Beispiel:

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

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

Hierdurch werden Zeilen zurückgegeben, in denen die Werte von Zellencode nach dem zweiten Vorkommen von 'X' am Ende 'AAA' enthalten.

PLUS

Syntax

Daten PLUS Summand; Daten + Summand

Parameter

Daten

Der Zellenbereich mit Zahlen, zu denen etwas addiert werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Summand

Die Zahl(en), die zu allen Werten in der angegebenen Spalte addiert werden soll(en). Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Daten2 muss mit der Anzahl der Spalten in Daten1 übereinstimmen, es sei denn, bei Daten2 handelt es sich um eine Konstante. Informationen zur Formatdefinition von Summand (identisch mit Daten) finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

PLUS addiert die Werte in den beiden angegebenen Datenbereichen. PLUS gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils zu der entsprechenden Spalte von Daten die entsprechende Spalte von Summand addiert wird (d. h., zur ersten Spalte von Daten wird die erste Spalte von Summand addiert, zur zweiten Spalte die zweite Spalte usw.).

Wenn es sich bei Summand um eine Konstante handelt, wird jeder Wert in Daten um den betreffenden Wert erhöht. Wenn Summand eine oder mehrere Spalten enthält, werden die Berechnungen zeilenweise zwischen einer Spalte von Daten und einer Spalte von Summand ausgeführt. Zur ersten Zeile von Daten wird die erste Zeile von Summand addiert, zur zweiten Zeile die zweite Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Der Operator PLUS kann durch ein Pluszeichen (+) abgekürzt werden.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

```
TEMP = 3 PLUS 4 oder TEMP = 3 + 4
```

Erstellt eine neue Spalte TEMP, die den Wert sieben enthält.

TEMP = W1 + 8
Erstellt eine neue Spalte TEMP, in der jeder Wert den Inhalt von Spalte W1 plus acht darstellt.
TEMP = W1 + W1
Erstellt eine neue Spalte TEMP, die jeweils den zweifachen Inhalt von Spalte W1 enthält.
TEMP = W1 + V2
Erstellt eine neue Spalte TEMP, wobei jeder Wert den Zeilenwert von Spalte W1 plus den entsprechenden Zeilenwert von Spalte W2 darstellt.
TEMP = W1:W3 + W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. Die Spalte TEMP enthält die Werte von Spalte W1 plus die entsprechenden Zeilenwerte von Spalte W4. Die Spalte WX addiert Spalte W2 und Spalte W5. Die Spalte WY addiert Spalte W3 und Spalte W6.
TEMP = W1[10:20] + W2 oder TEMP = W1[10:20] + W2[1:11]
Erstellt eine neue Spalte TEMP, deren erste 11 Zellen das Ergebnis der Addition der Werte in Zeile 10-20 von Spalte W1 und der Werte in Zeile 1-11 von Spalte W2 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
MINUS	Subtrahiert einen angegebenen Datenbereich von einem anderen.
SUM oder TOTAL	Berechnet die Summe eines Zellenbereichs.

POW

Syntax

Basis POW Exponent; Basis ^ Exponent

Parameter

Basis

Die numerischen Werte, die in die Potenz eines Exponenten erhoben werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Basis (identisch mit Daten) finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Exponent

Die exponentielle(n) Zahl(en), in deren Potenz die Werte in Basis erhoben werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Exponent muss mit der Anzahl der Spalten in Basis übereinstimmen, es sei denn, bei Exponent handelt es sich um eine Konstante. Informationen zur Formatdefinition von Exponent (identisch mit Daten) finden Sie im

Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

POW erhebt die Werte im ersten Datenbereich in die im zweiten Datenbereich angegebene Potenz (berechnet wird also $\text{base}^{\text{exponent}}$). POW gibt für jede Eingabespalte eine neue Spalte zurück, die jeweils das Ergebnis der Erhebung von Basis in die Potenz von Exponent enthält (d. h., die erste Spalte von Basis wird in die Potenz der ersten Spalte von Exponent erhoben, die zweite Spalte in die der zweiten Spalte usw.).

Wenn es sich bei Exponent um eine Konstante handelt, wird jeder Wert in Basis in die betreffende Potenz erhoben. Wenn Exponent eine oder mehrere Spalten enthält, werden die Berechnungen zeilenweise zwischen einer Spalte von Basis und einer Spalte von Exponent ausgeführt. Die erste Zeile von Basis wird in die Potenz der ersten Zeile von Exponent erhoben, die zweite Zeile in die der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Der Operator POW kann durch einen Zirkumflex (^) abgekürzt werden. Beispielsweise ist $\text{TEMP} = 2^8$ äquivalent zu $\text{TEMP} = 2 \text{ POW } 8$.

Anmerkung: Wenn der Wert x zu groß oder zu klein ist, wird ein Überlauffehler zurückgegeben. Dies ist der Fall, wenn $\text{Basis}^{\text{Exponent}}$ den maximalen oder minimalen 32-Bit-Gleitkommawert überschreitet.

Beispiele

$\text{TEMP} = 2 \text{ POW } 3$ oder $\text{TEMP} = 2^3$
Erstellt eine neue Spalte TEMP, die den Wert acht enthält.
$\text{TEMP} = \text{W1} \wedge 0,5$
Erstellt eine neue Spalte TEMP, in der jeder Wert die Quadratwurzel des Inhalts von Spalte W1 darstellt (äquivalent zu $\text{SQRT}(\text{W1})$).
$\text{TEMP} = \text{W1} \wedge \text{W3}$
Erstellt eine neue Spalte TEMP, wobei jeder Zeilenwert von Spalte W1 in die Potenz des entsprechenden Zeilenwerts von Spalte W2 erhoben wird.
$\text{TEMP} = \text{W1}:\text{W3} \wedge \text{W4}:\text{W6}$
Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Werte von Spalte W1 in die Potenz der entsprechenden Zeilenwerten von Spalte W4 erhoben. Die Spalte WX enthält das Ergebnis der Erhebung von Spalte W2 in die durch die entsprechenden Werte von Spalte W5 angegebene Potenz. Die Spalte WY enthält das Ergebnis der Erhebung von Spalte W3 of die durch die entsprechenden Werte von Spalte W6 angegebene Potenz.
$\text{TEMP} = \text{W1}[10:20] \text{ POW } \text{W2}$ oder $\text{TEMP} = \text{W1}[10:20] \text{ POW } \text{W2}[1:11]$
Erstellt eine neue Spalte TEMP, deren erste 11 Zellen das Ergebnis der Erhebung der Werte in Zeile 10-20 von Spalte W1 in die durch die Werte in Zeile 1-10 von Spalte W2 angegebene Potenz enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
EXP	Erhebt die natürliche Zahl (e) in die Potenz, die durch den Inhalt jeder Zelle im angegebenen Datenbereich angegeben ist.
LN oder LOG	Berechnet den natürlichen Logarithmus des Inhalts des angegebenen Datenbereichs.
LN2	Berechnet den binären Logarithmus des Inhalts des angegebenen Datenbereichs.
LN10	Berechnet den dekadischen Logarithmus des Inhalts des angegebenen Datenbereichs.

RANDOM

Syntax

RANDOM(Anzahl [, Seed]); RANDOM(Anzahl, Wert1, Wert2 [, Seed])

Parameter

Anzahl

Die Anzahl der Zufallszahlen, die generiert werden sollen. Dieser Wert muss eine positive Ganzzahl größer als null sein.

Wert1

Eine Grenze für die Zufallszahlen, die generiert werden sollen. Dabei kann es sich um einen konstanten Wert oder um einen Ausdruck handeln, dessen Auswertung eine Konstante ergibt. Wird dieser Parameter nicht angegeben, ist der Standard null.

Wert2

Die andere Grenze für die Zufallszahlen, die generiert werden sollen. Dabei kann es sich um einen konstanten Wert oder um einen Ausdruck handeln, dessen Auswertung eine Konstante ergibt. Wird dieser Parameter nicht angegeben, ist der Standard eins.

Seed

Ein optionaler Startwert für die Generierung von Zufallszahlen. Der Wert muss eine ganze Zahl sein.

Beschreibung

RANDOM generiert eine Spalte von Zufallszahlen. Dabei wird eine neue Spalte mit Anzahl Zufallszahlen zurückgegeben. Wenn Wert1 und Wert2 angegeben sind, werden die Zufallszahlen zwischen diesen Grenzen (inklusive) generiert. Wenn sie nicht angegeben sind, werden standardmäßig Werte zwischen null und eins generiert. Wenn Seed angegeben ist, wird dieser Wert als Startwert für die Generierung von Zufallszahlen verwendet.

Anmerkung: Wenn Seed größer-gleich 2^{32} ist, wird der Wert durch $2^{32}-1$ ersetzt. Werte von Seed über 2^{24} werden gerundet (d. h., die Genauigkeit geht verloren). Deshalb können sich bei demselben Wert von Seed mehrere Werte ergeben.

Beispiele

TEMP = RANDOM()
Erstellt eine neue Spalte TEMP, die Zufallszahlen mit unbegrenzter Länge enthält.
TEMP = RANDOM(100)
Erstellt eine neue Spalte TEMP, die 100 Zufallszahlen zwischen 0,0 und 1,0 enthält.
TEMP = RANDOM(100, 5943049)
Erstellt eine neue Spalte TEMP mit 100 Zufallszahlen, die ausgehend von dem Startwert 5943049 generiert werden.
TEMP = RANDOM(100, 0, 100)
Erstellt eine neue Spalte TEMP, die 100 Zufallszahlen zwischen 0 und 100,0 enthält.
TEMP = RANDOM(100, 0, 100, 5943049)
Erstellt eine neue Spalte TEMP mit 100 Zufallszahlen zwischen 0 und 100, die ausgehend von dem Startwert 5943049 generiert werden.

Zugehörige Funktionen

Funktion	Beschreibung
RANDOM_GAUSS	Gibt die angegebene Anzahl von Zufallswerten aus einer gaußschen Verteilung zurück.
SAMPLE_RANDOM	Gibt Spalte(n) von n Zellen zurück, die jeweils eine Zufallsstichprobe aus dem angegebenen Datenbereich enthalten.

RANDOM_GAUSS

Syntax

RANDOM_GAUSS(Anzahl [, Seed]); RANDOM_GAUSS(Anzahl, Mittelwert, Stdabw [, Seed])

Parameter

Anzahl

Die Anzahl der Zufallszahlen, die generiert werden sollen. Dieser Wert muss eine positive Ganzzahl größer als null sein.

Mittelwert

Der Mittelwert der gaußschen Kurve. Dabei kann es sich um einen konstanten Wert oder um einen Ausdruck handeln, dessen Auswertung eine Konstante ergibt. Wird dieser Parameter nicht angegeben, ist der Standard null.

Stdabw

Die Standardabweichung der gaußschen Kurve. Dabei kann es sich um einen konstanten Wert oder um einen Ausdruck handeln, dessen Auswertung eine Konstante ergibt. Wird dieser Parameter nicht angegeben, ist der Standard eins.

Seed

Ein optionaler Startwert für die Generierung von Zufallszahlen. Der Wert muss eine ganze Zahl sein. (Wenn der angegebene Wert keine Ganzzahl ist, wird er automatisch durch die Untergrenze des Werts ersetzt.)

Beschreibung

RANDOM_GAUSS generiert eine Spalte von Zufallszahlen, der eine gaußsche Verteilung zugrunde liegt. Dabei wird eine neue Spalte mit Anzahl Zufallszahlen zurückgegeben. Wenn Mittelwert und Stdabw angegeben sind, wird bei der Generierung der Zufallszahlen eine gaußsche Verteilung mit diesem Mittelwert und dieser Standardabweichung zugrunde gelegt. Wenn sie nicht angegeben sind, hat die gaußsche Verteilung standardmäßig den Mittelwert null und die Standardabweichung eins. Wenn Seed angegeben ist, wird dieser Wert als Startwert für die Generierung von Zufallszahlen verwendet.

Beispiele

TEMP = RANDOM_GAUSS(100)
Erstellt eine neue Spalte TEMP mit 100 Werten, die zufällig aus einer gaußschen Verteilung mit Mittelwert null und Einheitsstandardabweichung gewonnen wurden.
TEMP = RANDOM_GAUSS(500, 3)
Erstellt eine neue Spalte TEMP mit 100 Werten, die zufällig aus einer gaußschen Verteilung mit Mittelwert null und Einheitsstandardabweichung gewonnen wurden. Die Zahl 3 wird als Generierungswert für den Zufallsgenerator verwendet.
TEMP = RANDOM_GAUSS(5000, 100, 32)
Erstellt eine neue Spalte TEMP mit 5000 Werten, die zufällig aus einer gaußschen Verteilung mit Mittelwert 100 und Standardabweichung 32 gewonnen wurden.
TEMP = RANDOM_GAUSS(500, -1, 2, 3)
Erstellt eine neue Spalte TEMP mit 500 Werten, die zufällig aus einer gaußschen Verteilung mit Mittelwert -1 und Standardabweichung 2 gewonnen wurden. Die Zahl 3 wird als Generierungswert für den Zufallsgenerator verwendet.

Zugehörige Funktionen

Funktion	Beschreibung
RANDOM	Gibt die angegebene Anzahl von Zufallszahlen zurück.
SAMPLE_RANDOM	Gibt Spalte(n) von n Zellen zurück, die jeweils eine Zufallsstichprobe aus dem angegebenen Datenbereich enthalten.

RANK



Syntax

RANK(Daten [, n-Bins])

Parameter

Daten

Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt.

n-Bins

Die Anzahl der Bins, in die Daten unterteilt werden soll. Der Standardwert ist zehn.

Beschreibung

RANK unterteilt Daten in n-Bins (Standard 10) Gruppen, die jeweils die etwa gleiche Anzahl unterschiedlicher Werte enthalten, und gibt die Gruppe zurück, der die jeweilige Datenzeile angehört. Die Ausgabe umfasst 1 bis n-Bins Werte; wenn die Anzahl unterschiedlicher Werte von Daten kleiner als n-Bins ist, liegt die Anzahl der ausgegebenen Werte zwischen 1 und der Anzahl unterschiedlicher Werte von Daten.

RANK legt für den Wert von n-Bins automatisch eine Obergrenze von 1024*1024 fest. Dies ist auch die maximale Anzahl unterschiedlicher Werte, die verfolgt werden sollen; nachfolgende unterschiedliche Werte werden in den höchsten Bin aufgenommen.

Beispiele

```
TEMP = RANK(W6)
```

Erstellt eine neue Spalte TEMP, wobei jeder Wert, von 1 bis 10, den Bin von Daten darstellt, dem die Zeile angehört. In diesem Fall wird die Standardzahl von Bins, 10, zugrunde gelegt.

```
TEMP = RANK(W6, 15)
```

Erstellt eine neue Spalte TEMP, wobei jeder Wert, von 1 bis 15, den Bin von Daten darstellt, dem die Zeile angehört.

```
TEMP = REPEAT(3, W1:W3)
```

Erstellt drei neue Spalten TEMP, WX und WY. TEMP enthält drei Kopien von Spalte W1, WX enthält drei Kopien von Spalte W2 und WY enthält drei Kopien von Spalte W3. Wenn die Spalten W1 bis W3 unterschiedlich lang sind, werden die kürzeren Spalten bis zur Länge der längsten Spalte aufgefüllt.

```
TEMP = REPEAT(10, W1[10:50]:W2)
```

Erstellt zwei neue Spalten TEMP und WX. Die Werte in der Spalte TEMP sind 10 Kopien von Zeile 10-50 von Spalte W1, die Werte in der Spalte WX sind 10 Kopien von Zeile 10-50 von Spalte W2.

```
TEMP = REPEAT((1,2,3), (10, 20, 30))
```

Erstellt eine neue Spalte TEMP, die die Zellenwerte 10, 20, 20, 30, 30, 30 enthält.

```
TEMP = REPEAT(W1, W2)
```

Erstellt eine neue Spalte TEMP. Der Wert in Zelle W2[1] wird W1[1] Mal wiederholt, der Wert in Zelle W2[2] wird W1[2] Mal wiederholt usw., bis zum Ende von Spalte W1.

```
TEMP = REPEAT(W1, W2:W3)
```

Erstellt zwei neue Spalten TEMP und WX. Die Spalte TEMP enthält Kopien von Zellen in W2; die Spalte WX enthält Kopien von Zellen in W3. Es gibt W1[1] Kopien von W2[1] und W3[1], W1[2] Kopien von W2[2] und W3[2]. Dies setzt sich fort bis zum Ende von Spalte W1 oder zum Ende der längsten Spalte in Daten, je nachdem, welche Spalte kürzer ist. Kürzere Spalten in Daten werden mit Nullen aufgefüllt.

```
TEMP = REPEAT(3, W1, COL)
```

Erstellt drei neue Spalten TEMP, WX und WY, die jeweils eine Kopie von Spalte W1 darstellen.

```
TEMP = REPEAT(2, W1:W2, COL)
```

Erstellt vier neue Spalten TEMP, WX, WY und WZ. Spalte TEMP ist eine Kopie von Spalte W1, Spalte WX ist eine Kopie von Spalte W2, Spalte WY ist eine Kopie von Spalte W1 und Spalte WZ ist eine Kopie von Spalte W2.

REPEAT



Syntax

```
REPEAT(Anzahl, Daten [, Schlüsselwort])
```

Parameter

Anzahl

Die Anzahl der Wiederholungen des angegebenen Datenbereichs. Dabei kann es sich um eine Konstante, eine Spalte oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Alle Werte müssen positive Ganzzahlen sein.

Daten

Die numerischen Werte, die wiederholt werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie der angegebene Datenbereich repliziert wird. Wählen Sie eine der folgenden Angaben aus:

ROW - Wiederholt Zeilen von Daten vertikal (Standard)

COL - Wiederholt Spalten von Daten horizontal

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

REPEAT wiederholt den angegebenen Datenbereich eine Anzahl von Malen vertikal (COL) oder horizontal (ROW).

Bei dem Schlüsselwort COL wird für jede Eingabespalte eine neue Spalte zurückgegeben, die Anzahl Kopien von Daten enthält, die vertikal verkettet sind. Wenn die Eingabespalten von Daten unterschiedlich lang sind, werden die kürzeren Spalten automatisch aufgefüllt. Numerische Spalten werden mit Nullen aufgefüllt, Zeichenfolgespalten werden mit leeren Zeichenfolgen (") aufgefüllt.

Bei dem Schlüsselwort ROW hängt die Anzahl der zurückgegebenen Spalten davon ab, ob es sich bei Anzahl um eine Konstante oder eine Spalte handelt. Wenn Anzahl eine Konstante ist, gibt REPEAT Anzahl Mal die Anzahl der Spalten in Daten zurück. Wenn Anzahl eine Spalte ist, gibt REPEAT die angegebene Anzahl von Kopien von jeder Spalte von Daten zurück. Dabei gibt der erste Zeilenwert von Anzahl an, wie oft die erste Spalte von Daten wiederholt werden soll, der zweite Zeilenwert gibt an, wie oft die zweite Spalte von Daten wiederholt werden soll, usw. Zeilen, die größer als die Anzahl der Spalten in Daten sind, werden ignoriert.

Beispiele

TEMP = REPEAT(10, 1) oder TEMP = REPEAT(10, 1, ROW)
Erstellt eine neue Spalte TEMP, die zehnmal den Wert 1 enthält.
TEMP = REPEAT(2, W1)
Erstellt eine neue Spalte TEMP, die zwei Kopien von Spalte W1 enthält, die vertikal verkettet sind.
TEMP = REPEAT(3, W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. TEMP enthält drei Kopien von Spalte W1, WX enthält drei Kopien von Spalte W2 und WY enthält drei Kopien von Spalte W3. Wenn die Spalten W1 bis W3 unterschiedlich lang sind, werden die kürzeren Spalten bis zur Länge der längsten Spalte aufgefüllt.
TEMP = REPEAT(10, W1[10:50]:W2)
Erstellt zwei neue Spalten TEMP und WX. Die Werte in der Spalte TEMP sind 10 Kopien von Zeile 10-50 von Spalte W1, die Werte in der Spalte WX sind 10 Kopien von Zeile 10-50 von Spalte W2.
TEMP = REPEAT((1,2,3), (10, 20, 30))
Erstellt eine neue Spalte TEMP, die die Zellenwerte 10, 20, 20, 30, 30, 30 enthält.
TEMP = REPEAT(W1, W2)
Erstellt eine neue Spalte TEMP. Der Wert in Zelle W2[1] wird W1[1] Mal wiederholt, der Wert in Zelle W2[2] wird W1[2] Mal wiederholt usw., bis zum Ende von Spalte W1.

```
TEMP = REPEAT(W1, W2:W3)
```

Erstellt zwei neue Spalten TEMP und WX. Die Spalte TEMP enthält Kopien von Zellen in W2; die Spalte WX enthält Kopien von Zellen in W3. Es gibt W1[1] Kopien von W2[1] und W3[1], W1[2] Kopien von W2[2] und W3[2]. Dies setzt sich fort bis zum Ende von Spalte W1 oder zum Ende der längsten Spalte in Daten, je nachdem, welche Spalte kürzer ist. Kürzere Spalten in Daten werden mit Nullen aufgefüllt.

```
TEMP = REPEAT(3, W1, COL)
```

Erstellt drei neue Spalten TEMP, WX und WY, die jeweils eine Kopie von Spalte W1 darstellen.

```
TEMP = REPEAT(2, W1:W2, COL)
```

Erstellt vier neue Spalten TEMP, WX, WY und WZ. Spalte TEMP ist eine Kopie von Spalte W1, Spalte WX ist eine Kopie von Spalte W2, Spalte WY ist eine Kopie von Spalte W1 und Spalte WZ ist eine Kopie von Spalte W2.

ROTATE_LEFT



Syntax

```
ROTATE_LEFT(Spaltenzahl, Daten)
```

Parameter

Spaltenzahl

Die Anzahl der Spalten, die nach links rotiert werden sollen. Dieser Wert muss eine nicht negative Ganzzahl sein. Bei dem Wert null werden die Zeilen ohne Rotation kopiert.

Daten

Die numerischen Werte, die nach links rotiert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

ROTATE_LEFT rotiert den angegebenen Datenbereich um eine Anzahl von Spalten nach links. ROTATE_LEFT gibt für jede Eingabespalte eine neue Spalte zurück, die eine Kopie der entsprechenden Eingabespalte enthält, die um Spaltenzahl Positionen nach links rotiert worden ist. Spalten, die über den linken Rand geschoben werden, werden auf der rechten Seite wieder angefügt.

Anmerkung: ROTATE_LEFT verarbeitet nur numerische Daten. Unter den im Parameter Daten enthaltenen Daten darf sich kein ASCII-Text befinden.

Beispiele

<pre>TEMP = ROTATE_LEFT(1, MERGE(1, 2, 3))</pre>
Erstellt drei neue Spalten TEMP, WX und WY. TEMP enthält den Wert 2, WX enthält den Wert 3 und WY enthält den Wert 1.
<pre>TEMP = ROTATE_LEFT(0, W1:W3)</pre>
Erstellt drei neue Spalten TEMP, WX und WY. Spalte TEMP ist eine Kopie von Spalte W1, Spalte WX ist eine Kopie von Spalte W2 und Spalte WY ist eine Kopie von Spalte W3.
<pre>TEMP = ROTATE_LEFT(4, W1:W3)</pre>
Erstellt drei neue Spalten TEMP, WX und WY. Spalte TEMP ist eine Kopie von Spalte W2, Spalte WX ist eine Kopie von Spalte W3 und Spalte WY ist eine Kopie von Spalte W1.
<pre>TEMP = ROTATE_LEFT(1, W1[10:50]:W2)</pre>
Erstellt zwei neue Spalten TEMP und WX, die jeweils in den ersten 41 Zeilen Werte enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind die Werte in Zeile 10-50 von Spalte W2, die Werte in der Spalte WX sind die Werte in Zeile 10-50 von Spalte W1.

Zugehörige Funktionen

Funktion	Beschreibung
ROTATE_RIGHT	Rotiert die Spalten im angegebenen Datenbereich nach rechts.

ROTATE_RIGHT



Syntax

ROTATE_RIGHT(Spaltenzahl, Daten)

Parameter

Spaltenzahl

Die Anzahl der Spalten, die nach rechts rotiert werden sollen. Dieser Wert muss eine nicht negative Ganzzahl sein. Bei dem Wert null werden die Zeilen ohne Rotation kopiert.

Daten

Die numerischen Werte, die nach rechts rotiert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

ROTATE_RIGHT rotiert den angegebenen Datenbereich um eine Anzahl von Spalten nach rechts. ROTATE_RIGHT gibt für jede Eingabespalte eine neue Spalte zurück, die eine Kopie der entsprechenden Eingabespalte enthält, die um Spaltenzahl Positio-

nen nach rechts rotiert worden ist. Spalten, die über den rechten Rand geschoben werden, werden auf der linken Seite wieder angefügt.

Anmerkung: ROTATE_RIGHT verarbeitet nur numerische Daten. Unter den im Parameter Daten enthaltenen Daten darf sich kein ASCII-Text befinden.

Beispiele

TEMP = ROTATE_RIGHT(1, MERGE(1, 2, 3))
Erstellt drei neue Spalten TEMP, WX und WY. TEMP enthält den Wert 3, WX enthält den Wert 1 und WY enthält den Wert 2.
TEMP = ROTATE_RIGHT(0, W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Spalte TEMP ist eine Kopie von Spalte W1, Spalte WX ist eine Kopie von Spalte W2 und Spalte WY ist eine Kopie von Spalte W3.
TEMP = ROTATE_RIGHT(4, W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Spalte TEMP ist eine Kopie von Spalte W3, Spalte WX ist eine Kopie von Spalte W1 und Spalte WY ist eine Kopie von Spalte W2.
TEMP = ROTATE_RIGHT(1, W1[10:50]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils in den ersten 41 Zeilen Werte enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind die Werte in Zeile 10-50 von Spalte W2, die Werte in der Spalte WX sind die Werte in Zeile 10-50 von Spalte W1.

Zugehörige Funktionen

Funktion	Beschreibung
ROTATE_LEFT	Rotiert die Spalten im angegebenen Datenbereich nach links.

ROUND

Syntax

ROUND(Daten)

Parameter

Daten

Die numerischen Werte, die gerundet werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

ROUND rundet die Werte im angegebenen Datenbereich auf die nächste Ganzzahl. Dabei wird für jede Eingabespalte eine neue Spalte zurückgegeben, die den gerundeten Wert der Zahlen in der entsprechenden Eingabespalte enthält. Genau in der Mitte liegende Zahlen werden aufgerundet (so wird 2,5 zu 3,0 und -2,5 zu -2,0 gerundet).

Beispiele

TEMP = ROUND(3.2)
Erstellt eine neue Spalte TEMP, die den Wert drei enthält.
TEMP = ROUND(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert den gerundeten Wert des Inhalts von Spalte W1 darstellt.
TEMP = ROUND(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die gerundeten Werte des Inhalts von Spalte W1, die Werte von Spalte WX sind die gerundeten Werte des Inhalts von Spalte W2 und die Werte von Spalte WY sind die gerundeten Werte des Inhalts von Spalte W3.
TEMP = ROUND(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen die gerundeten Werte in Zeile 10-20 von Spalte W1 enthalten. Die anderen Zellen in TEMP sind leer.
TEMP = ROUND(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind die gerundeten Werte der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind die gerundeten Werte der entsprechenden Zeilen von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
INT	Berechnet den abgerundeten Ganzzahlwert des Inhalts des angegebenen Datenbereichs.
MOD	Berechnet den Modulo-Wert des Inhalts des angegebenen Datenbereichs.
TRUNCATE	Berechnet den Ganzzahlanteil jedes Werts im angegebenen Datenbereich.

ROWNUM

Syntax

ROWNUM()

Beschreibung

ROWNUM generiert fortlaufende Zahlen von eins bis zur Anzahl der Einträge. Die Zahl für den ersten Datensatz ist eins, die Zahl für den zweiten Datensatz ist zwei usw.

Anmerkung: ROWNUM kann maximal zwei Milliarden Einträge verarbeiten.

RTRIM

Syntax

RTRIM(Daten)

Parameter

Daten

Beschreibung

RTRIM entfernt nachfolgende Leerzeichen aus jedem Zeichenfolgewert im angegebenen Datenbereich und gibt die konvertierte Zeichenfolge zurück. RTRIM gibt für jede Eingabespalte eine neue Spalte zurück.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

```
Temp = RTRIM "Gold "
```

Erstellt eine neue Zeichenfolge Temp, die "Gold" enthält.

SAMPLE_RANDOM



Syntax

```
SAMPLE_RANDOM(Anzahl, Daten [, Seed])
```

Parameter

Anzahl

Die Anzahl der Stichproben, die aus jeder Spalte im angegebenen Datenbereich genommen werden sollen.

Daten

Die Werte, denen Zufallsstichproben entnommen werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Seed

Ein optionaler Startwert für die Generierung von Zufallszahlen. Der Wert muss eine ganze Zahl sein. (Wenn der angegebene Wert keine Ganzzahl ist, wird er automatisch durch die Untergrenze des Werts ersetzt.)

Beschreibung

SAMPLE_RANDOM erstellt Zufallsstichproben für den angegebenen Datenbereich. SAMPLE_RANDOM gibt für jede Eingabespalte eine neue Spalte mit Anzahl Zahlen zurück, die als Zufallsstichproben der entsprechenden Eingabespalte von Daten entnommen wurden. Die Stichproben werden in der Reihenfolge genommen, in der sie in der jeweiligen Spalte auftreten (d. h., die relative Reihenfolge der Datenwerte bleibt erhalten). Wenn Seed angegeben ist, wird dieser Wert als Startwert für die

Generierung von Zufallszahlen verwendet.

Beispiele

TEMP = SAMPLE_RANDOM(100, 3)
Erstellt eine neue Spalte TEMP mit 100 Zellen, die alle den Wert 3 enthalten.
TEMP = SAMPLE_RANDOM(100, W1)
Erstellt eine neue Spalte TEMP mit 100 Werten, in der jeder Wert eine Zufallsstichprobe des Inhalts von Spalte W1 darstellt.
TEMP = SAMPLE_RANDOM(50, W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils 50 Werte enthalten. Die Werte in der Spalte TEMP sind Zufallsstichproben aus Spalte W1, die Werte von Spalte WX sind Zufallsstichproben aus Spalte W2 und die Werte von Spalte WY sind Zufallsstichproben aus Spalte W3.
TEMP = SAMPLE_RANDOM(100, W1[10:50]:W3)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils 41 Werte enthalten. Die Werte in der Spalte TEMP sind Zufallsstichproben aus Zelle 10-50 von Spalte W1, die Werte von Spalte WX sind Zufallsstichproben aus Zelle 10-50 von Spalte W2 und die Werte von Spalte WY sind Zufallsstichproben aus Zelle 10-50 von Spalte W3. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
RANDOM	Gibt die angegebene Anzahl von Zufallszahlen zurück.
RANDOM_GAUSS	Gibt die angegebene Anzahl von Zufallswerten aus einer gaußschen Verteilung zurück.
SUBSAMPLE	Reduziert Daten durch Rückgabe jedes n-ten Zeilenwerts.

SELECT



Syntax

```
SELECT(Spaltennummern, Daten); SELECT(Anfangsspalte, Daten);  
SELECT(Anfangsspalte, Endspalte, Daten)
```

Parameter

Spaltennummern

Der Name einer Spalte, die die Spaltennummern für die Extraktion aus dem angegebenen Datenbereich enthält (beispielsweise würden bei einer Spalte mit den Nummern 1, 3, 4 und 7 die erste, die dritte, die vierte und die siebte Spalte aus dem angegebenen Datenbereich extrahiert). Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von

Spaltennummern (identisch mit Daten) finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Anfangsspalte

Die numerische Position der Spalte, die aus dem angegebenen Datenbereich extrahiert werden soll. Bei dem Wert eins wird die erste Spalte des angegebenen Datenbereichs extrahiert.

Endspalte

Wenn dieser Parameter angegeben wird, dient Anfangsspalte als Ausgangspunkt und muss eine Spalte oder einen Zellenbereich darstellen. Der Endpunkt wird durch Endspalte angegeben. Dieser Wert muss größer als Anfangsspalte sein.

Daten

Der Zellenbereich, der die zu extrahierende(n) Spalte(n) enthält. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

SELECT gibt die angegebenen Spalten aus einem Datenbereich zurück. Die vorgesehenen Spalten können auf verschiedene Weise angegeben werden. Der Parameter Spaltennummern enthält die Nummern der Spalten, die aus dem angegebenen Datenbereich extrahiert werden sollen. Der Parameter Anfangsspalte ermöglicht die Extraktion einer einzelnen Spalte. Geben Sie zum Extrahieren einer fortlaufenden Spaltenfolge Anfangsspalte als Ausgangspunkt (muss eine Spalte oder ein Zellenbereich sein) und Endspalte als Endspalte an.

Wenn Daten skalar (d. h. eine Konstante oder eine Variable mit einem Einzelwert) ist, wird bei Auswahl der ersten Spalte eine neue Spalte zurückgegeben, die diese Konstante enthält. Wenn bei einer Konstante eine andere Spalte ausgewählt wird, wird eine Spalte mit ??? zurückgegeben.

Dieses Makro wird häufig in komplexere Funktionen integriert.

Anmerkung: Um mehrere nicht benachbarte Spalten zu extrahieren, können Sie mit der Makrofunktion COLUMN eine Spalte erstellen, die die vorgesehenen Spaltennummern enthält. Vergleichen Sie dazu die folgenden Beispiele.

Beispiele

TEMP = SELECT(1, 3)Erstellt eine neue Spalte TEMP, die die Zahl drei enthält.
TEMP = SELECT(1, W1) oder TEMP = SELECT(1, W1:W3) Erstellt eine neue Spalte TEMP, die eine Kopie von Spalte W1 darstellt.
TEMP = SELECT(2, 4, W1:W5) Erstellt drei neue Spalten TEMP, WX und WY. Spalte TEMP ist eine Kopie von Spalte W2, Spalte WX ist eine Kopie von Spalte W3 und Spalte WY ist eine Kopie von Spalte W4.

TEMP = SELECT(COLUMN(1,4), W6:W10)
Erstellt zwei neue Spalten TEMP und WX. TEMP ist eine Kopie von Spalte W6 und WX ist eine Kopie von Spalte W9.
TEMP = SELECT(COLUMN(1,4), W6[25:74]:W10)
Erstellt zwei neue Spalten TEMP und WX, die jeweils 50 Werte enthalten. TEMP ist eine Kopie der Zellen 25-74 von Spalte W6 und WX ist eine Kopie der Zellen 25-74 von Spalte W9. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
EXTRACT	Extrahiert Zeilen aufgrund der Werte in einer Prädikatspalte.
MERGE	Erstellt eine Datengruppe durch horizontale Verkettung der Eingabewerte.

SIGN

Syntax

SIGN(Daten)

Parameter

Daten

Die numerischen Werte, deren Vorzeichen berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

SIGN testet das Vorzeichen der Werte im angegebenen Datenbereich. SIGN gibt für jede Eingabespalte eine neue Spalte zurück, die das Vorzeichen der Zahlen in der entsprechenden Eingabespalte enthält. Bei allen Werten größer als null wird eine positive Eins zurückgegeben; bei allen Werten kleiner als null wird eine negative Eins zurückgegeben; bei allen Werten gleich null wird eine Null zurückgegeben.

Beispiele

TEMP = SIGN(-3)
Erstellt eine neue Spalte TEMP, die den Wert -1 enthält.
TEMP = SIGN(MERGE(3, -2, 0))
Erstellt drei neue Spalten TEMP, WX und WY. TEMP enthält den Wert 1, WX enthält den Wert -1 und WY enthält den Wert 0.

TEMP = SIGN(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert das Vorzeichen des Inhalts von Spalte W1 darstellt.
TEMP = SIGN(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die Vorzeichen des Inhalts von Spalte W1, die Werte von Spalte WX sind die Vorzeichen des Inhalts von Spalte W2 und die Werte von Spalte WY sind die Vorzeichen des Inhalts von Spalte W3.
TEMP = SIGN(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils die Vorzeichen der Werte in Zeile 10-20 von Spalte W1 enthalten. Die anderen Zellen in TEMP sind leer.
TEMP = SIGN(W1[10:50]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-41 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind die Vorzeichen der Werte in Zeile 10-50 von Spalte W1, die Werte in der Spalte WX sind die Vorzeichen der Werte in Zeile 10-50 von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
ABS	Berechnet den absoluten Wert des Inhalts des angegebenen Datenbereichs.
INVERSE	Berechnet den Negativwert des Inhalts des angegebenen Datenbereichs.

SIN

Syntax

SIN(Daten [, Einheit])

Parameter

Daten

Die numerischen Werte, deren Sinus berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Einheit

Dieses optionale Schlüsselwort legt fest, ob die Eingabewerte und Ergebnisse als Grad oder als Radiant interpretiert werden. Wählen Sie eine der folgenden Angaben aus:

RADIAN - Führt die Berechnungen in Radiant aus (Standard)

DEGREE - Führt die Berechnungen in Grad aus

Wird dieser Parameter nicht angegeben, ist der Standard Radiant. (Die Umrechnung von Radiant in Grad erfolgt durch Division durch Pi und Multiplikation mit 180.)

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

SIN berechnet den Sinus der Werte im angegebenen Datenbereich. SIN gibt für jede Eingabespalte eine neue Spalte zurück, die den Sinus der Zahlen in der entsprechenden Eingabespalte enthält.

Beispiele

TEMP = SIN(PI/2) oder TEMP = SIN(PI/2, 0) oder TEMP = SIGN(PI/2, RADIAN)
Erstellt eine neue Spalte TEMP, die den Wert eins enthält.
TEMP = SIN(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert den Sinus (in Radiant) des Inhalts von Spalte W1 darstellt.
TEMP = SIN(W1:W3, 1) oder TEMP = SIN(W1:W3, DEGREE)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der Sinus des Inhalts von Spalte W1, die Werte von Spalte WX sind der Sinus des Inhalts von Spalte W2 und die Werte von Spalte WY sind der Sinus des Inhalts von Spalte W3. Alle Werte sind in Grad angegeben.
TEMP = SIN(W1[10:50]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-41 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der Sinus der Werte in Zeile 10-50 von Spalte W1, die Werte in der Spalte WX sind der Sinus der Werte in Zeile 10-50 von Spalte W2. Alle Werte sind in Radiant angegeben.

Zugehörige Funktionen

Funktion	Beschreibung
ASIN	Berechnet den Arkussinus des Inhalts des angegebenen Datenbereichs.
COS	Berechnet den Kosinus des Inhalts des angegebenen Datenbereichs.
SINH	Berechnet den Hyperbelsinus des Inhalts des angegebenen Datenbereichs.
TAN	Berechnet den Tangens des Inhalts des angegebenen Datenbereichs.

Syntax

SINH(Daten [, Einheit])

Parameter

Daten

Die numerischen Werte, deren Hyperbelsinus berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Einheit

Dieses optionale Schlüsselwort legt fest, ob die Eingabewerte und Ergebnisse als Grad oder als Radiant interpretiert werden. Wählen Sie eine der folgenden Angaben aus:

RADIAN - Führt die Berechnungen in Radiant aus (Standard)

DEGREE - Führt die Berechnungen in Grad aus

Wird dieser Parameter nicht angegeben, ist der Standard Radiant. (Die Umrechnung von Radiant in Grad erfolgt durch Division durch Pi und Multiplikation mit 180.)

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

SINH berechnet den Hyperbelsinus der Werte im angegebenen Datenbereich. SINH gibt für jede Eingabespalte eine neue Spalte zurück, die den Hyperbelsinus der Zahlen in der entsprechenden Eingabespalte enthält. Bei x in Radiant ist der Hyperbelsinus einer Zahl

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

Dabei ist e die natürliche Zahl, 2,7182818.

Anmerkung: Wenn der Wert x zu groß ist, wird ein Überlauferfehler zurückgegeben. Dies geschieht, wenn $\sinh(x)$ den maximalen 32-Bit-Gleitkommawert überschreitet.

Beispiele

$TEMP = \text{SINH}(1)$ oder $TEMP = \text{SINH}(1, 0)$ oder $TEMP = \text{SINH}(1, \text{RADIAN})$
Erstellt eine neue Spalte TEMP, die den Wert 1,18 enthält.
$TEMP = \text{SINH}(W1)$
Erstellt eine neue Spalte TEMP, in der jeder Wert den Hyperbelsinus (in Radiant) des Inhalts von Spalte W1 darstellt.
$TEMP = \text{SINH}(W1:W3, 1)$ oder $TEMP = \text{SINH}(W1:W3, \text{DEGREE})$
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der Hyperbelsinus des Inhalts von Spalte W1, die Werte von Spalte WX sind der Hyperbelsinus des Inhalts von Spalte W2 und die Werte von Spalte WY sind der Hyperbelsinus des Inhalts von Spalte W3. Alle Werte sind in Grad angegeben.
$TEMP = \text{SINH}(W1[10:50]:W2)$
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-41 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der Hyperbelsinus der Werte in Zeile 10-50 von Spalte W1, die Werte in der Spalte WX sind der Hyperbelsinus der Werte in Zeile 10-50 von Spalte W2. Alle Werte sind in Radiant angegeben.

Zugehörige Funktionen

Funktion	Beschreibung
COSH	Berechnet den Hyperbelkosinus des Inhalts des angegebenen Datenbereichs.
SIN	Berechnet den Sinus des Inhalts des angegebenen Datenbereichs.
TANH	Berechnet den Hyperbeltangens des Inhalts des angegebenen Datenbereichs.

SKEW



Syntax

$\text{SKEW}(\text{Daten } [, \text{Schlüsselwort}])$

Parameter

Daten

Die numerischen Werte, deren Schiefe berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt. In Daten müssen mindestens drei Werte enthalten sein.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabedatenbereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

SKEW berechnet die Schiefe der Verteilung aller Zellen im angegebenen Datenbereich. Die Schiefe misst den Grad der Asymmetrie einer Verteilung bezüglich ihres Mittelwerts. Ein positiver Schiefenwert zeigt eine Verteilung mit einer asymmetrischen Flanke an, die zu positiveren Werten hin geneigt ist; eine negative Schiefe zeigt eine Verteilung mit einer asymmetrischen Flanke an, die zu negativeren Werten hin geneigt ist. Ein Schiefenwert null zeigt an, dass die Verteilung bezüglich ihres Mittelwerts symmetrisch ist.

Die Schiefe wird wie folgt berechnet:

$$\frac{n}{(n-1)(n-2)} \sum_1^n \left(\frac{x_i - \text{mean}}{\sigma} \right)^3$$

Dabei steht n für die Anzahl der Stichproben in der Verteilung, *Mittelwert* für den Durchschnitt und σ für die Standardabweichung der Verteilung. Es müssen mindestens drei Datenwerte angegeben werden, um die Schiefe berechnen zu können.

Anmerkung: Wenn die Standardabweichung $\sigma = 0$ ist, gibt SKEW null zurück.

Beispiele

TEMP = SKEW(3) oder TEMP = SKEW(3, ALL)
Erstellt eine neue Spalte TEMP, die den Wert null enthält.
TEMP = SKEW(MERGE(3, 7, 2))
Erstellt eine neue Spalte TEMP, die den Wert 0,14 enthält.
TEMP = SKEW(W1)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Schiefe des Inhalts von Spalte W1 darstellt.

TEMP = SKEW(W1:W3)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Schiefe des Inhalts der Spalten W1, W2 und W3 darstellt.
TEMP = SKEW(W1[10:20])
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Schiefe der Zellen in Zeile 10-20 von Spalte W1 darstellt.
TEMP = SKEW(W1[1:5]:W4)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Schiefe der Zellen in Zeile 1-5 von Spalte W1 bis W4 darstellt.
TEMP = SKEW(W1:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY. Der Einzelwert in der Spalte TEMP ist die Schiefe des Inhalts von Spalte W1, der Einzelwert in der Spalte WX ist die Schiefe des Inhalts von Spalte W2 und der Einzelwert in der Spalte WY ist die Schiefe des Inhalts von Spalte W3.
TEMP = SKEW(W1[1:5]:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils einen Einzelwert enthalten. Der Wert in der Spalte TEMP ist die Schiefe der Zellen in Zeile 1-5 von Spalte W1, der Wert in der Spalte WX ist die Schiefe der Zellen in Zeile 1-5 von Spalte W2 und der Wert in der Spalte WY ist die Schiefe der Zellen in Zeile 1-5 von Spalte W3.
TEMP = SKEW(W1:W3, ROW)
Erstellt eine neue Spalte TEMP, in der jeder Zelleneintrag die Schiefe der entsprechenden Zeile in Spalte W1, W2 und W3 darstellt.
TEMP = SKEW(W1[10:50]:W3, ROW)
Erstellt eine neue Spalte TEMP, in der die ersten 41 Zellen die Schiefe der Werte in Zeile 10-50 von Spalte W1 bis W3 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
AVG oder MEAN	Berechnet das arithmetische Mittel oder den Durchschnitt eines Zellenbereichs.
KURTOSIS	Berechnet die Kurtosis eines Zellenbereichs.
STAT	Berechnet das erste bis vierte Moment des angegebenen Datenbereichs.
VARIANCE	Berechnet die Varianz eines Zellenbereichs.

SLIDE_WINDOW



Syntax

SLIDE_WINDOW(Breite, Daten [, Inkrement])

Parameter

Breite

Die Größe (vertikale Zeilenzahl) des gleitenden Fensters.

Daten

Der Zellenbereich, über den ein Fenster geschoben werden soll, um Daten zu generieren. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makro-funktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Inkrement

Die Anzahl der Zeilen, um die das Fenster jedes Mal verschoben werden soll; der Standardwert ist eins. Es muss sich um eine Ganzzahl größer als null (und kleiner als die Länge von Daten) handeln.

Beschreibung

SLIDE_WINDOW schiebt ein Fenster mit der angegebenen Größe über den angegebenen Datenbereich und erstellt Muster aus den Datenwerten im Fenster. Dabei werden (Anzahl der Eingabespalten x Breite) Spalten zurückgegeben. Das gleitende Fenster beginnt am Anfang von Daten und bedeckt Breite Zeilen. Die Datenwerte in diesem Fenster (von links nach rechts und von oben nach unten zu lesen) werden im Ausgabedatenbereich zu einer einzelnen Zeile verkettet. Das gleitende Fenster wird jedes Mal um Inkrement Zeilen verschoben.

Nehmen wir beispielsweise an, dass die Spalten W1 und W2 die folgenden Daten enthalten:

```
10 2 20 3 30 4 40 5 50 ...
```

Dann erzeugt der Ausdruck W3 = SLIDE_WINDOW(2, W1:W2) in den Spalten W3:W6 die folgende Ausgabe:

```
10 2 20 2 20 3 30 3 30 4 40 ...
```

Die ersten beiden Zeilen ergeben die erste Ausgabezeile. Anschließend wird das Fenster eine Zeile nach unten geschoben, um das nächste Muster zu erstellen, usw.

Anmerkung: Diese Funktion kann zum Erstellen von Mustern aus Zeitseriendaten eingesetzt werden.

Beispiele

```
TEMP = SLIDE_WINDOW(1, W1)
```

Erstellt eine neue Spalte TEMP, die eine Kopie der Werte in Spalte W1 enthält.

```
TEMP = SLIDE_WINDOW(3, W1:W3)
```

Erstellt neun neue Spalten, wobei jede Zeile ein Fenster mit drei mal drei Datenfeldern aus den Spalten W1:W3 enthält. Die Zeilen 1-3 der Eingabe bilden die erste Ausgabezeile, die Zeilen 2-4 die zweite usw.

TEMP = SLIDE_WINDOW(2, W1:W3[10:20])
Erstellt sechs neue Spalten, wobei jede Zeile ein Fenster mit drei mal zwei Datenfeldern aus den Zeilen 10-20 der Spalten W1:W3 enthält. Die Zeilen 10-11 der Eingabe bilden die erste Ausgabezeile, die Zeilen 11-12 die zweite usw.
TEMP = SLIDE_WINDOW(2, MERGE(W1, W3, W5))
Erstellt sechs neue Spalten, wobei jede Zeile ein Fenster mit drei mal zwei Datenfeldern aus den Spalten W1, W3 und W5 enthält. Die Zeilen 1-2 der Eingabe bilden die erste Ausgabezeile, die Zeilen 2-3 die zweite usw.
TEMP = SLIDE_WINDOW(1, W1:W3, 2)
Erstellt drei neue Spalten; dabei enthält die erste Zeile Daten aus W1[1]:W3, die zweite Zeile Daten aus W1[3]:W3, die dritte Zeile Daten aus W1[5]:W3 usw. (jede andere Zeile wird übersprungen).
TEMP = SLIDE_WINDOW(10, W1, 10)
Erstellt zehn neue Spalten; dabei enthält die erste Zeile Daten aus W1[1:10], die zweite Zeile Daten aus W1[11:20], die dritte Zeile Daten aus W1[21:30] usw.
TEMP = SLIDE_WINDOW(3, W1:W2, 5)
Erstellt sechs neue Spalten, wobei jede Zeile ein Fenster mit zwei mal drei Datenfeldern aus den Spalten W1:W2 enthält. Die erste Zeile enthält Daten aus W1[1:3]:W2, die zweite Zeile enthält Daten aus W1[6:8]:W2, die dritte Zeile enthält Daten aus W1[11:13]:W2 usw.

Zugehörige Funktionen

Funktion	Beschreibung
GRID	Gibt ein Grid aller möglichen Wertekombinationen (eine pro Zeile) zurück.

SORT



Syntax

SORT(Spalte [, Schlüsselwort]); SORT(Spalte, Daten [, Schlüsselwort])

Parameter

Spalte

Im ersten Format (ohne Angabe von Daten) ist dies die Spalte mit den zu sortierenden Daten (numerisch oder Text). Dabei kann es sich um eine Konstante, eine Spalte, einen einspaltigen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Dieser Datenbereich darf nicht mehr als 2^{29} Werte enthalten.

Daten

Wenn dieser Parameter angegeben wird, sind dies die zu sortierenden Daten, wobei Spalte die Sortierkriterien enthält (Daten kann Spalten mit numerischen Daten und Text enthalten). Bei dem Parameter Daten kann es sich um eine Spalte, einen

Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt. Dieser Datenbereich darf nicht mehr als ²²⁹ Zeilen enthalten.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, ob die Werte in aufsteigender (Minimum zu Maximum) oder absteigender (Maximum zu Minimum) Reihenfolge sortiert werden sollen. Wählen Sie eine der folgenden Angaben aus:

ASCEND - Daten in aufsteigender (zunehmender) Reihenfolge sortieren (Standard)

DESCEND - Daten in absteigender (abnehmender) Reihenfolge sortieren

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

SORT sortiert die Werte im angegebenen Datenbereich (entweder Spalte oder Daten anhand von Spalte) gemäß dem Wert von Schlüsselwort (ASCEND oder DESCEND). SORT gibt für jede zu sortierende Eingabespalte eine neue Spalte zurück. Wenn nur Spalte angegeben ist, werden die Werte in Spalte je nach dem Parameter Schlüsselwort in aufsteigender oder absteigender Reihenfolge sortiert. Bei einer Textspalte bedeutet aufsteigend die alphabetische Reihenfolge (a-z) und absteigend die umgekehrte Reihenfolge (z-a). Wenn Daten ebenfalls angegeben ist, richtet sich die Sortierung nach den in Spalte enthaltenen Sortierkriterien.

Anmerkung: Wenn für Spalte ein einspaltiger Zellenbereich angegeben wird, um die *entsprechenden* Zeilen von Daten zu sortieren, müssen Sie für Daten denselben Zellenbereich angeben. Andernfalls werden standardmäßig die ersten *n* Zeilen von Daten sortiert. Zum Sortieren der entsprechenden Zeilen können Sie beispielsweise Folgendes angeben: TEMP = SORT(W1[100:200], W2[100:200]:W5). Andernfalls ist TEMP = SORT(W1[100:200], W2:W5) äquivalent zu TEMP = SORT(W1[100:200], W2[1:101]:W5).

Beispiele

TEMP = SORT(COLUMN(5, 3, 2, 4, 1)) oder TEMP = SORT(COLUMN(5, 3, 2, 4, 1), ASCEND)
Erstellt eine neue Spalte TEMP, die die Werte 1, 2, 3, 4 und 5 enthält.
TEMP = SORT(COLUMN("b", "c", "a"))
Erstellt eine neue Spalte TEMP, die die Zeichenfolgen a, b und c enthält.
TEMP = SORT(10...15, DESCEND)
Erstellt eine neue Spalte TEMP, die die Werte 15, 14, 13, 12, 11 und 10 enthält.
TEMP = SORT(W1)
Erstellt eine neue Spalte TEMP, in der die Werte aus Spalte W1 aufsteigend sortiert sind.

<p>TEMP = SORT(W1, W1:W3)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die aufsteigend sortierten Werte aus Spalte W1. Die Werte der Spalte WX sind der entsprechende Inhalt von Spalte W2, die Werte der Spalte WY sind der entsprechende Inhalt von Spalte W3.</p>
<p>TEMP = SORT(W1[10:20], DESCEND)</p> <p>Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen die absteigend sortierten Werte der Zellen in Zeile 10-20 von Spalte W1 enthalten. Die anderen Zellen in TEMP sind leer.</p>
<p>TEMP = SORT(W1[5:10], W2) oder TEMP = SORT(W1[5:10], W2[1:6])</p> <p>Erstellt eine neue Spalte TEMP, deren erste 6 Zellen die Werte aus Zeile 1-6 von Spalte W2 enthalten, die in absteigender Reihenfolge der Zellen 5-10 von Spalte W1 sortiert sind. Die anderen Zellen in TEMP sind leer.</p>
<p>TEMP = SORT(W1[5:10], W2[5:10])</p> <p>Erstellt eine neue Spalte TEMP, deren erste 6 Zellen die sortierten Werte der Zellen in Zeile 5-10 von Spalte W2 enthalten, die gemäß der absteigenden Reihenfolge der Zellen 5-10 von Spalte W1 sortiert sind. Die anderen Zellen in TEMP sind leer.</p>
<p>TEMP = SORT(W1[10:50], W2:W3) oder TEMP = SORT(W1[10:50], W2[1:41]:W3)</p> <p>Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-41 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind die Werte aus Zeile 1-41 von Spalte W2, die gemäß Zeile 10-50 von Spalte W1 sortiert sind. Die Werte in der Spalte WX sind die Werte aus Zeile 1-41 von Spalte W3, die gemäß Zeile 10-50 von Spalte W1 sortiert sind. Spalte W1 wird in aufsteigender Reihenfolge sortiert.</p>

SQRT

Syntax

SQRT(Daten)

Parameter

Daten

Die numerischen Werte, deren Quadratwurzel berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

SQRT berechnet die Quadratwurzel der Werte im angegebenen Datenbereich. SQRT gibt für jede Eingabespalte eine neue Spalte zurück, die die positive Quadratwurzel der Zahlen in der entsprechenden Eingabespalte enthält.

Anmerkung: If a value in the defined data range is negative, a ??? für die Zelle zurückgegeben.

Beispiele

TEMP = SQRT(2)
Erstellt eine neue Spalte TEMP, die den Wert 1,41 enthält.
TEMP = SQRT(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert die Quadratwurzel des Inhalts von Spalte W1 darstellt.
TEMP = SQRT(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils die Quadratwurzeln des Inhalts von Spalte W1, die Werte von Spalte WX sind die Quadratwurzeln des Inhalts von Spalte W2 und die Werte von Spalte WY sind die Quadratwurzeln des Inhalts von Spalte W3.
TEMP = SQRT(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils die Quadratwurzeln der Werte in Zeile 10-20 von Spalte W1 enthalten. Die anderen Zellen in TEMP sind leer.
TEMP = SQRT(W1[10:50]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-41 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind die Quadratwurzeln der Werte in Zeile 10-50 von Spalte W1, die Werte in der Spalte WX sind die Quadratwurzeln der Werte in Zeile 10-50 von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
DIV	Dividiert einen angegebenen Datenbereich durch einen anderen.
MULT	Multipliziert die Inhalte zweier Datenbereiche.
POW	Erhebt einen Basiswert in die Potenz des oder der angegebenen Exponenten.

STAT



Syntax

STAT(Daten [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, deren Momente (d. h. Durchschnitt, Standardabweichung, Schiefe und Kurtosis) berechnet werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von

Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt. In Daten müssen mindestens drei Werte enthalten sein.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabedatenbereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

STAT berechnet die ersten vier Momente der Werte im angegebenen Datenbereich. Das erste Moment ist der Durchschnitt. Das zweite Moment ist die Standardabweichung. Das dritte Moment ist die Schiefe und das vierte Moment ist die Kurtosis.

Die Anzahl der von der Makrofunktion STAT zurückgegebenen Spalten hängt von Schlüsselwort sowie der Anzahl der Spalten in Daten ab.

- Wenn das Schlüsselwort ALL verwendet wird (Standard), werden die Momente für alle Werte in Daten berechnet. Es wird eine Spalte zurückgegeben, die vier Werte enthält.
- Wenn das Schlüsselwort COL verwendet wird, werden die Momente für jede Eingabespalte gesondert berechnet. Für jede Eingabespalte wird eine Spalte zurückgegeben, die vier Werte enthält.
- Wenn das Schlüsselwort ROW verwendet wird, werden die Momente für jede Zeile von Daten gesondert berechnet. STAT gibt vier Spalten zurück. Die Momente werden für jede Zeile des Eingabedatenbereichs aufgelistet.

Beispiele

<code>TEMP = STAT(MERGE(1, 2, 3, 4, 5))</code> oder <code>TEMP = STAT(MERGE(1, 2, 3, 4, 5), ALL)</code>
Erstellt eine neue Spalte TEMP, die die Werte 3, 1,58, 0 und -1,2 enthält.
<code>TEMP = STAT(W1)</code>
Erstellt eine neue Spalte TEMP, die die ersten vier Momente von Spalte W1 enthält.
<code>TEMP = STAT(W1:W3)</code>
Erstellt eine neue Spalte TEMP, die die ersten vier Momente der Spalten W1, W2 und W3 enthält.
<code>TEMP = STAT(W1[10:20])</code>
Erstellt eine neue Spalte TEMP, die die ersten vier Momente der Zellen in Zeile 10-20 von Spalte W1 enthält.

<p>TEMP = STAT(W1[1:5]:W4)</p> <p>Erstellt eine neue Spalte TEMP, die die ersten vier Momente der Zellen in Zeile 1-5 von Spalte W1 bis W4 enthält.</p>
<p>TEMP = STAT(W1:W3, COL)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY. Die vier Werte in der Spalte TEMP sind die Momente von Spalte W1, die vier Werte von Spalte WX sind die Momente von Spalte W2 und die vier Werte von Spalte WY sind die Momente von Spalte W3.</p>
<p>TEMP = STAT(W1[1:5]:W3, COL) oder TEMP = STAT(W1[1:5]:W3[1:5], COL)</p> <p>Erstellt drei neue Spalten TEMP, WX und WY, die jeweils vier Werte enthalten. Die Werte in der Spalte TEMP sind die Momente der Zellen in Zeile 1-5 von Spalte W1, die Werte in der Spalte WX sind die Momente der Zellen in Zeile 1-5 von Spalte W2 und die Werte in der Spalte WY sind die Momente der Zellen in Zeile 1-5 von Spalte W3.</p>
<p>TEMP = STAT(W1:W3, ROW)</p> <p>Erstellt vier neue Spalten TEMP, WX, WY und WZ. TEMP enthält den Durchschnitt jeder Zeile in den Spalten W1, W2 und W3, WX enthält die Standardabweichung, WY enthält die Schiefe und WZ enthält die Kurtosis.</p>
<p>TEMP = STAT(W1[50:100]:W3, ROW) oder TEMP = STAT(W1[50:100]:W3[50:100], ROW)</p> <p>Erstellt vier neue Spalten TEMP, WX, WY und WZ, die jeweils 51 Zeilen enthalten. TEMP enthält den Durchschnitt, WX enthält die Standardabweichung, WY enthält die Schiefe und WZ enthält die Kurtosis. Die erste Zeile entspricht Zeile 50 in den Spalten W1, W2 und W3. Die zweite Zeile entspricht Zeile 51 usw.</p>

Zugehörige Funktionen

Funktion	Beschreibung
AVG oder MEAN	Berechnet das arithmetische Mittel oder den Durchschnitt eines Zellenbereichs.
KURTOSIS	Berechnet die Kurtosis eines Zellenbereichs.
SKEW	Berechnet die Schiefe der Verteilung eines Zellenbereichs.
STDV oder STDEV	Berechnet die Standardabweichung eines Zellenbereichs.
VARIANCE	Berechnet die Varianz eines Zellenbereichs.

STDV oder STDEV

Syntax

STDV(Daten [, Schlüsselwort]); STDEV(Daten [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, deren Standardabweichung berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabedatenbereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Anmerkung: Viele Makrofunktionen setzen die Schlüsselwortparameter {ALL | COL | ROW} ein. Diese Schlüsselwörter gelten nicht für **IBM Kampagne**, da es sich bei den Eingabedaten immer um eine Einzelspalte oder ein Einzelfeld handelt. Das Makro verhält sich immer so, als ob das Schlüsselwort COL angegeben würde. Deshalb brauchen Sie diese Schlüsselwörter nicht anzugeben, wenn Sie **IBM Kampagne** verwenden.

Beschreibung

STDV berechnet die Standardabweichung aller Zellen im angegebenen Datenbereich. Die Standardabweichung einer Verteilung ist die Quadratwurzel der Varianz. Die Standardabweichung wird wie folgt berechnet:

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

Dabei steht x für eine Stichprobe, n für die Anzahl der Stichproben und *Mittelwert* für den Durchschnitt der Verteilung.

Anmerkung: Wenn die Anzahl der Stichproben $n = 1$ ist, gibt STDV einen Fehler zurück.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

```
TEMP = STDV(MERGE(1, 2, 1, 0)) oder TEMP = STDEV(MERGE(1, 2, 1, 0))
```

Erstellt eine neue Spalte TEMP, die den Wert 0,71 enthält.

TEMP = STDV(W1)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Standardabweichung des Inhalts von Spalte W1 darstellt.
TEMP = STDV(W1:W3)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Standardabweichung des Inhalts der Spalten W1, W2 und W3 darstellt.
TEMP = STDV(W1[1:5]:W4)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Standardabweichung der Zellen in Zeile 1-5 von Spalte W1 bis W4 darstellt.
TEMP = STDV(W1:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY. Der Einzelwert in der Spalte TEMP ist die Standardabweichung des Inhalts von Spalte W1, der Einzelwert in der Spalte WX ist die Standardabweichung des Inhalts von Spalte W2 und der Einzelwert in der Spalte WY ist die Standardabweichung des Inhalts von Spalte W3.
TEMP = STDV(W1[10:50]:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils einen Einzelwert enthalten. Der Wert in der Spalte TEMP ist die Standardabweichung der Zellen in Zeile 10-50 von Spalte W1, der Wert in der Spalte WX ist die Standardabweichung der Zellen in Zeile 10-50 von Spalte W2 und der Wert in der Spalte WY ist die Standardabweichung der Zellen in Zeile 10-50 von Spalte W3.
TEMP = STDV(W1:W3, ROW)
Erstellt eine neue Spalte TEMP, in der jeder Zelleneintrag die Standardabweichung der entsprechenden Zeile in Spalte W1, W2 und W3 darstellt.
TEMP = STDV(W1[1:5]:W3, ROW)
Erstellt eine neue Spalte TEMP, in der die Zellen in Zeile 1-5 die Standardabweichung der entsprechenden Zeile in Spalte W1 bis W3 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
KURTOSIS	Berechnet die Kurtosis eines Zellenbereichs.
SKEW	Berechnet die Schiefe der Verteilung eines Zellenbereichs.
STAT	Berechnet das erste bis vierte Moment des angegebenen Datenbereichs.
VAR	Berechnet die Varianz eines Zellenbereichs.

STRING_CONCAT

Syntax

STRING_CONCAT(Zeichenfolge1, Zeichenfolge2, ... ZeichenfolgeN)

Parameter

Zeichenfolge

Eine ASCII-Zeichenfolge, die verkettet werden soll. Dabei kann es sich um ASCII-Text in Anführungszeichen, eine Textspalte, einen Zellenbereich mit Text oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Der Abschnitt *Makrofunktionsparameter* im Kapitel *Verwendung von Makros* für Ihr Produkt enthält Informationen zur Formatdefinition von Zeichenfolge (identisch mit der von Daten).

Beschreibung

STRING_CONCAT verkettet die ASCII-Textwerte in den angegebenen Datenbereichen. Dabei wird für jede Eingabespalte eine neue Spalte zurückgegeben, die die verketteten Zeichenfolgen aus den entsprechenden Zeilen von Zeichenfolge enthält. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Die Gesamtbreite jeder Ergebniszeichenfolge darf 255 Zeichen nicht überschreiten.

Dieses Makro ist in IBM Interact verfügbar.

IBM Interact unterstützt auch die folgende Syntax:

```
STRING_CONCAT( Zeichenfolge1 , Zeichenfolge2 , ... ZeichenfolgeN )
```

Zum Beispiel ist STRING_CONCAT('a', 'b', 'c', 'd') gültig.

Beispiele

TEMP = STRING_CONCAT("HAUS", "BOOT")
Erstellt eine neue Spalte TEMP, die die ASCII-Zeichenfolge "HAUSBOOT" enthält.
TEMP = STRING_CONCAT(W1, ".")
Erstellt eine neue Spalte TEMP, in der jede Zeile die ASCII-Zeichenfolge in der entsprechenden Zeile von Spalte W1 sowie einen angehängten Punkt enthält.
TEMP = STRING_CONCAT(W1, W2)
Erstellt eine neue Spalte TEMP, wobei in jeder Zeile die ASCII-Zeichenfolge in Spalte W1 mit der Zeichenfolge in Spalte W2 verkettet ist.
TEMP = STRING_CONCAT(W1:W3, W4:W6)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die verketteten Zeichenfolgen aus den entsprechenden Zeilen der Spalten W1 und W4, die Werte der Spalte WX sind die verketteten Zeichenfolgen aus den entsprechenden Zeilen der Spalten W2 und W5 und die Werte der Spalte WY sind die verketteten Zeichenfolgen aus den entsprechenden Zeilen der Spalten W3 und W6.
TEMP = STRING_CONCAT(W1[5:10]:W2, W3:W4)
Erstellt zwei neue Spalten TEMP und WX. Die Werte in der Spalte TEMP sind Zeichenfolgen aus den Zeilen 5-10 von Spalte W1, die mit den Zeichenfolgen aus den Zeilen 1-6 von Spalte W3 verkettet sind. Die Werte in der Spalte WX sind Zeichenfolgen aus den Zeilen 5-10 von Spalte W2, die mit den Zeichenfolgen aus den Zeilen 1-6 von Spalte W4 verkettet sind.
TEMP = STRING_CONCAT('a', 'b', 'c', 'd')
Erstellt eine neue Spalte TEMP, die die ASCII-Zeichenfolge "abcd" enthält.

Zugehörige Funktionen

Funktion	Beschreibung
STRING_HEAD	Gibt die ersten n Zeichen jeder Zeichenfolge im angegebenen Datenbereich zurück.
STRING_LENGTH	Gibt die Länge jeder Zeichenfolge im angegebenen Datenbereich zurück.
STRING_SEG	Gibt das Zeichenfolgesegment zwischen zwei angegebenen Indizes zurück.
STRING_TAIL	Gibt die letzten n Zeichen jeder Zeichenfolge im angegebenen Datenbereich zurück.

STRING_HEAD

Syntax

STRING_HEAD(Zeichenzahl, Daten)

Parameter

Zeichenzahl

Die Anzahl der Zeichen, die vom Anfang jeder Zeichenfolge in Daten zurückgegeben werden sollen. Dieser Wert muss eine positive Ganzzahl größer als null sein.

Daten

ASCII-Zeichenfolgewerte. Dabei kann es sich um ASCII-Text in Anführungszeichen, eine Textspalte, einen Zellenbereich mit Text oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

STRING_HEAD gibt die ersten Zeichenzahl Zeichen jedes Zeichenfolgewerts im angegebenen Datenbereich zurück. Wenn Zeichenzahl größer als die Anzahl der Zeichen in einer Zeichenfolge ist, werden die verbleibenden Zeichen mit dem Nullzeichen "\0" aufgefüllt.

Beispiele

TEMP = STRING_HEAD(3, "JAN 1997")
Erstellt eine neue Spalte TEMP, die die ASCII-Zeichenfolge "JAN" enthält.
TEMP = STRING_HEAD(10, "Druck")
Erstellt eine neue Spalte TEMP, die die ASCII-Zeichenfolge "Druck" enthält.
TEMP = STRING_HEAD(5, W1)
Erstellt eine neue Spalte TEMP, die die ersten fünf Zeichen jeder Zeichenfolge in der Spalte W1 enthält.

<pre>TEMP = STRING_HEAD(1, W1:W3)</pre> <p>Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils das erste Zeichen der Zeichenfolgen in den entsprechenden Zeilen von Spalte W1, die Werte von Spalte WX sind das erste Zeichen der Zeichenfolgen in den entsprechenden Zeilen von Spalte W2 und die Werte von Spalte WY sind das erste Zeichen der Zeichenfolgen in den entsprechenden Zeilen von Spalte W3.</p>
<pre>TEMP = STRING_HEAD(12, W4[1:50]:W6]</pre> <p>Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die ersten 12 Zeichen der Zeichenfolgen in Zeile 1-50 von Spalte W1, die Werte von Spalte WX sind die ersten 12 Zeichen der Zeichenfolgen in Zeile 1-50 von Spalte W2 und die Werte von Spalte WY sind die ersten 12 Zeichen der Zeichenfolgen in Zeile 1-50 von Spalte W3.</p>

Zugehörige Funktionen

Funktion	Beschreibung
STRING_CONCAT	Verkettet zwei Zeichenfolgen aus den angegebenen Datenbereichen.
STRING_LENGTH	Gibt die Länge jeder Zeichenfolge im angegebenen Datenbereich zurück.
STRING_SEG	Gibt das Zeichenfolgesegment zwischen zwei angegebenen Indizes zurück.
STRING_TAIL	Gibt die letzten n Zeichen jeder Zeichenfolge im angegebenen Datenbereich zurück.

STRING_LENGTH

Syntax

STRING_LENGTH(Daten)

Parameter

Daten

ASCII-Zeichenfolgewerte, deren Länge berechnet werden soll. Dabei kann es sich um ASCII-Text in Anführungszeichen, eine Textspalte, einen Zellenbereich mit Text oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

STRING_LENGTH gibt die Länge jedes Zeichenfolgewerts im angegebenen Datenbereich zurück. Dabei wird für jede Eingabespalte eine neue Spalte zurückgegeben, die die Länge der entsprechenden Zeichenfolge enthält.

Anmerkung: Wenn STRING_LENGTH auf Spalten mit numerischen Daten angewendet wird, gibt die Funktion Nullen zurück.

Beispiele

<pre>TEMP = STRING_LENGTH("vier")</pre>
Erstellt eine neue Spalte TEMP, die den Wert 4 enthält.
<pre>TEMP = STRING_LENGTH(4)</pre>
Erstellt eine neue Spalte TEMP, die den Wert 0 enthält.
<pre>TEMP = STRING_LENGTH(W1)</pre>
Erstellt eine neue Spalte TEMP, in der jeder Wert die Länge der Zeichenfolge in der entsprechenden Zeile von Spalte W1 darstellt.
<pre>TEMP = STRING_LENGTH(W1:W3)</pre>
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die Längen der Zeichenfolgen in den entsprechenden Zeilen von Spalte W1, die Werte von Spalte WX sind die Längen der Zeichenfolgen in den entsprechenden Zeilen von Spalte W2 und die Werte von Spalte WY sind die Längen der Zeichenfolgen in den entsprechenden Zeilen von Spalte W3.
<pre>TEMP = STRING_LENGTH(W4[1:50]:W6]</pre>
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die Längen der Zeichenfolgen in Zeile 1-50 von Spalte W1, die Werte von Spalte WX sind die Längen der Zeichenfolgen in Zeile 1-50 von Spalte W2 und die Werte von Spalte WY sind die Längen der Zeichenfolgen in Zeile 1-50 von Spalte W3.

Zugehörige Funktionen

Funktion	Beschreibung
STRING_CONCAT	Verkettet zwei Zeichenfolgen aus den angegebenen Datenbereichen.
STRING_HEAD	Gibt die ersten n Zeichen jeder Zeichenfolge im angegebenen Datenbereich zurück.
STRING_SEG	Gibt das Zeichenfolgesegment zwischen zwei angegebenen Indizes zurück.
STRING_TAIL	Gibt die letzten n Zeichen jeder Zeichenfolge im angegebenen Datenbereich zurück.

STRING_PROPER

Syntax

STRING_PROPER(Daten)

Parameter

Daten

Der Zeichenfolgewart, der konvertiert werden soll.

Beschreibung

STRING_PROPER konvertiert jeden Zeichenfolgewart im angegebenen Datenbereich so, dass der erste Buchstabe oder irgendein Buchstabe, der auf ein Leerzeichen oder Symbol (außer dem Unterstrich) folgt, in einen Großbuchstaben und alle an-

deren Zeichen in Kleinbuchstaben geändert werden. `STRING_PROPER` gibt für jede Eingabespalte eine neue Spalte zurück, die die konvertierte Zeichenfolge der entsprechenden Eingabespalte enthält.

Beispiele

```
Temp = STRING_PROPER
```

STRING_SEG

Syntax

```
STRING_SEG(von, bis, Daten)
```

Parameter

von

Die Zeichenzahl ab dem Anfang der Zeichenfolge, bei der die Extraktion des Zeichenfolgesegments beginnen soll. Es muss sich um eine positive Ganzzahl größer als null und kleiner als `bis` handeln; andernfalls gibt `STRING_SEG` eine leere Zeichenfolge zurück.

bis

Die Zeichenzahl ab dem Anfang der Zeichenfolge, bei der die Extraktion des Zeichenfolgesegments enden soll. Es muss sich um eine positive Ganzzahl größer-gleich von handeln. Wenn `bis` gleich von ist (und `bis` kleiner-gleich der Länge der Zeichenfolge ist), wird ein Zeichen zurückgegeben.

Daten

ASCII-Zeichenfolgewerte. Dabei kann es sich um ASCII-Text in Anführungszeichen, eine Textspalte, einen Zellenbereich mit Text oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

`STRING_SEG` gibt aus jedem Zeichenfolgewart im angegebenen Datenbereich das Zeichenfolgesegment zwischen zwei Indizes zurück. Wenn von größer als die Länge einer Zeichenfolge ist, wird nichts zurückgegeben. Wenn bis größer als die Länge einer Zeichenfolge ist, werden alle Zeichen ab von zurückgegeben.

Beispiele

```
TEMP = STRING_SEG(1, 8, "15. JAN. 1997")
```

Erstellt eine neue Spalte TEMP, die die ASCII-Zeichenfolge "15. JAN." enthält.

```
TEMP = STRING_SEG(3, 20, "Druck")
```

Erstellt eine neue Spalte TEMP, die die ASCII-Zeichenfolge "uck" enthält.

TEMP = STRING_SEG(5, 6, W1)
Erstellt eine neue Spalte TEMP, die das fünfte und das sechste Zeichen jeder Zeichenfolge in der Spalte W1 enthält.
TEMP = STRING_SEG(10, 20, W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die Zeichen 10-20 der Zeichenfolgen in den entsprechenden Zeilen von Spalte W1, die Werte von Spalte WX sind die Zeichen 10-20 der Zeichenfolgen in den entsprechenden Zeilen von Spalte W2 und die Werte von Spalte WY sind die Zeichen 10-20 der Zeichenfolgen in den entsprechenden Zeilen von Spalte W3.
TEMP = STRING_SEG(5, 10, W4[1:50]:W6]
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die Zeichen 5-10 der Zeichenfolgen in Zeile 1-50 von Spalte W1, die Werte von Spalte WX sind die Zeichen 5-10 der Zeichenfolgen in Zeile 1-50 von Spalte W2 und die Werte von Spalte WY sind die Zeichen 5-10 der Zeichenfolgen in Zeile 1-50 von Spalte W3.

Zugehörige Funktionen

Funktion	Beschreibung
STRING_CONCAT	Verkettet zwei Zeichenfolgen aus den angegebenen Datenbereichen.
STRING_HEAD	Gibt die ersten n Zeichen jeder Zeichenfolge im angegebenen Datenbereich zurück.
STRING_LENGTH	Gibt die Länge jeder Zeichenfolge im angegebenen Datenbereich zurück.
STRING_TAIL	Gibt die letzten n Zeichen jeder Zeichenfolge im angegebenen Datenbereich zurück.

STRING_TAIL

Syntax

STRING_TAIL(Zeichenzahl, Daten)

Parameter

Zeichenzahl

Die Anzahl der Zeichen, die vom Ende jeder Zeichenfolge in Daten zurückgegeben werden sollen. Dieser Wert muss eine positive Ganzzahl größer als null sein.

Daten

ASCII-Zeichenfolgewerte. Dabei kann es sich um ASCII-Text in Anführungszeichen, eine Textspalte, einen Zellenbereich mit Text oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

STRING_TAIL gibt die letzten Zeichenzahl Zeichen jedes Zeichenfolgewerts im angegebenen Datenbereich zurück. Alle Zeichenfolgewerte werden bis zur Länge der längsten Zeichenfolge mit Nullzeichen "\0" aufgefüllt. Anschließend werden die letzten Zeichenzahl Zeichen jeder Zeichenfolge zurückgegeben. Wenn Zeichenzahl größer als die Anzahl der Zeichen in einer Zeichenfolge ist, wird die gesamte Zeichenfolge zurückgegeben.

Beispiele

<pre>TEMP = STRING_TAIL(3, "JAN 1997")</pre>
Erstellt eine neue Spalte TEMP, die die ASCII-Zeichenfolge "997" enthält.
<pre>TEMP = STRING_TAIL(10, "Druck")</pre>
Erstellt eine neue Spalte TEMP, die die ASCII-Zeichenfolge "Druck" enthält.
<pre>TEMP = STRING_TAIL(5, W1)</pre>
Erstellt eine neue Spalte TEMP, die die letzten fünf Zeichen jeder Zeichenfolge in der Spalte W1 enthält.
<pre>TEMP = STRING_TAIL(1, W1:W3)</pre>
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils das letzte Zeichen der Zeichenfolgen in den entsprechenden Zeilen von Spalte W1, die Werte von Spalte WX sind das letzte Zeichen der Zeichenfolgen in den entsprechenden Zeilen von Spalte W2 und die Werte von Spalte WY sind das letzte Zeichen der Zeichenfolgen in den entsprechenden Zeilen von Spalte W3.
<pre>TEMP = STRING_TAIL(12, W4[1:50]:W6]</pre>
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die letzten 12 Zeichen der Zeichenfolgen in Zeile 1-50 von Spalte W1, die Werte von Spalte WX sind die letzten 12 Zeichen der Zeichenfolgen in Zeile 1-50 von Spalte W2 und die Werte von Spalte WY sind die letzten 12 Zeichen der Zeichenfolgen in Zeile 1-50 von Spalte W3.

Zugehörige Funktionen

Funktion	Beschreibung
STRING_CONCAT	Verkettet zwei Zeichenfolgen aus den angegebenen Datenbereichen.
STRING_HEAD	Gibt die ersten n Zeichen jeder Zeichenfolge im angegebenen Datenbereich zurück.
STRING_LENGTH	Gibt die Länge jeder Zeichenfolge im angegebenen Datenbereich zurück.
STRING_SEG	Gibt das Zeichenfolgesegment zwischen zwei angegebenen Indizes zurück.

SUBSAMPLE



Syntax

SUBSAMPLE(Anzahl, Daten)

Parameter

Anzahl

Die Anzahl der Stichproben, die extrahiert werden sollen. Dabei muss es sich um eine positive Ganzzahl handeln, die kleiner als die Anzahl der Zellen im angegebenen Datenbereich ist (die Makrofunktion SUBSAMPLE kann also nicht dazu verwendet werden, die Anzahl der Dateneinträge durch Replikation zu *erhöhen*).

Daten

Die Werte, denen Stichproben entnommen werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

SUBSAMPLE erstellt gleichmäßig die angeforderte Anzahl von Stichproben aus dem angegebenen Datenbereich. SUBSAMPLE gibt für jede Eingabespalte eine neue Spalte zurück, die jeweils Anzahl Zeilen mit Zahlen enthält, die gleichmäßig aus der entsprechenden Eingabespalte extrahiert worden sind. Der erste Zeilenwert und jeder *n*-te folgende Zeilenwert werden zurückgegeben, sodass insgesamt Anzahl Werte extrahiert werden.

Anmerkung: Mit dieser Makrofunktion kann die Anzahl der Stichproben sowohl erhöht als auch verringert werden.

Beispiele

TEMP = SUBSAMPLE(100, W1)
Erstellt eine neue Spalte TEMP mit 100 Werten, die gleichmäßig aus der Spalte W1 extrahiert worden sind.
TEMP = SUBSAMPLE(50, W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils 50 Werte enthalten. Die Werte in der Spalte TEMP sind die Stichprobenwerte des Inhalts von Spalte W1, die Werte von Spalte WX sind die Stichprobenwerte des Inhalts von Spalte W2 und die Werte von Spalte WY sind die Stichprobenwerte des Inhalts von Spalte W3.
TEMP = SUBSAMPLE(5, W1[0:100])
Erstellt eine neue Spalte TEMP, die in den ersten fünf Zeilen Werte enthält. Die Daten werden gleichmäßig aus den Zeilen 0-100 von Spalte W1 extrahiert.
TEMP = SUBSAMPLE(250, W1[1:10]:W2)
Erstellt zwei neue Spalten TEMP und WX, die in den ersten 250 Zeilen Werte enthalten. Die Werte in der Spalte TEMP sind gleichmäßig aus Zeile 1-400 von Spalte W1 extrahiert, die Werte in der Spalte WX sind gleichmäßig aus Zeile 1-400 von Spalte W2 extrahiert.

Zugehörige Funktionen

Funktion	Beschreibung
EXTRACT	Extrahiert Zeilen aufgrund der Werte in einer Prädikatspalte.

Funktion	Beschreibung
SAMPLE_RANDOM	Gibt Spalte(n) von n Zellen zurück, die jeweils eine Zufallsstichprobe aus dem angegebenen Datenbereich enthalten.

SUBSTITUTE



Syntax

SUBSTITUTE(Daten, Quellentabelle, Zieltabelle)

Parameter

Daten

Die numerischen Werte oder Zeichenfolgewerte, die konvertiert werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Quellentabelle

Eine Spalte mit Werten, die konvertiert werden sollen. Die Länge der Spalte Quellentabelle muss mit der Länge von Zieltabelle übereinstimmen.

Zieltabelle

Eine Spalte mit den Werten, in die konvertiert werden soll. Die Länge der Spalte Zieltabelle muss mit der Länge von Quellentabelle übereinstimmen.

Beschreibung

SUBSTITUTE konvertiert Werte in Daten anhand der Ersetzungspaare, die in Quellentabelle und Zieltabelle angegeben sind. Jeder Wert in Daten, der in Quellentabelle gefunden wird, wird durch den Wert in der entsprechenden Zeile von Zieltabelle ersetzt.

SUBSTITUTE kann sowohl auf numerische Werte als auch auf Zeichenfolgewerte angewendet werden. Es wird immer ein Datenbereich zurückgegeben, dessen Dimensionen denen von Daten entsprechen.

Anmerkung: Wenn Sie mit SUBSTITUTE Zeichenfolgen in Zahlen umwandeln oder umgekehrt, müssen *alle* Werte in Daten konvertiert werden. Demzufolge muss jeder Wert aus Daten in Quellentabelle enthalten sein. Andernfalls enthält die Ergebniszeile ???.

Beispiele

<pre>TEMP = SUBSTITUTE(COLUMN(1,5,10), COLUMN(1), COLUMN(7))</pre> <p>Erstellt eine neue Spalte TEMP, die die Werte 7, 5, 10 enthält (der Wert 1 wird durch 7 ersetzt).</p>
<pre>TEMP = SUBSTITUTE(COLUMN("blau", "rot"), COLUMN("blau", "rot"), COLUMN(0, 1))</pre> <p>Erstellt eine neue Spalte TEMP, die die Werte 0 und 1 enthält (die Zeichenfolge "blau" wird durch 0 und "rot" wird durch 1 ersetzt).</p>
<pre>TEMP = SUBSTITUTE(W1, W2, W3)</pre> <p>Erstellt eine neue Spalte TEMP, die die Werte aus Spalte W1 enthält, wobei jeder in Spalte W2 gefundene Wert durch den Wert in der entsprechenden Zeile von Spalte W3 ersetzt wird.</p>
<pre>TEMP = SUBSTITUTE(W1:W2, W4, W5)</pre> <p>Erstellt zwei neue Spalten TEMP und WX, die die Werte aus Spalte W1 bzw. W2 enthalten, wobei jeder in Spalte W4 gefundene Wert durch den Wert in der entsprechenden Zeile von Spalte W5 ersetzt wird.</p>
<pre>TEMP = SUBSTITUTE(W1[10:20]:W2, W4, W5)</pre> <p>Erstellt zwei neue Spalten TEMP und WX, die die Werte aus Zeile 10-20 von Spalte W1 bzw. W2 enthalten, wobei jeder in Spalte W4 gefundene Wert durch den Wert in der entsprechenden Zeile von Spalte W5 ersetzt wird.</p>

Zugehörige Funktionen

Funktion	Beschreibung
EXTRACT	Extrahiert Zeilen aufgrund der Werte in einer Prädikatspalte.
ISMEMBER	Prüft einen Eingabebereich anhand einer "Tabelle" von Werten und gibt eine Eins zurück, wenn ein Wert in der Tabelle enthalten ist, andernfalls eine Null.

SUBSTR oder SUBSTRING

Syntax

SUBSTR(Zeichenfolgewart, Startpos[, nZeichen]) oder SUBSTR(Zeichenfolgewart FROM Startpos[FOR nZeichen]) SUBSTRING(Zeichenfolgewart, Startpos[, nZeichen]) oder SUBSTRING(Zeichenfolgewart FROM Startpos[FOR nZeichen])

Parameter

Zeichenfolgewart

Die Zeichenfolge, der eine Unterzeichenfolge entnommen werden soll.

Startpos

Das erste Zeichen, ab dem die Unterzeichenfolge extrahiert werden soll.

nZeichen

Die Anzahl der zu extrahierenden Zeichen (muss größer-gleich 0 sein). Wenn dieser Wert nicht angegeben ist, werden alle restlichen Zeichen in Zeichenfolgewert extrahiert.

Beschreibung

SUBSTR oder SUBSTRING extrahiert nZeichen Zeichen aus der Zeichenfolge, beginnend bei Startpos. Wenn nZeichen nicht angegeben ist, werden durch SUBSTR und SUBSTRING die Zeichen von Startpos bis zum Ende der Zeichenfolge extrahiert. Nachfolgende Leerzeichen werden automatisch abgeschnitten.

Dieses Makro ist in IBM Interact verfügbar.

Wichtig: IBM Interact unterstützt nur die folgenden Formate:
SUBSTR(Zeichenfolgewert, Startpos[, nZeichen]) oder
SUBSTRING(Zeichenfolgewert, Startpos[, nZeichen])

Beispiele

SUBSTR SUBSTR Rückgabe	("abcdef" FROM 1 FOR 2) ("abcdef",1,2) 'ab'
SUBSTR SUBSTR Rückgabe	("abcdef" FROM -2 FOR 4) ("abcdef",-2,4) 'a'
SUBSTR SUBSTR Rückgabe	("abcdef" FROM 3) ("abcdef",3) 'cdef'

SUM

Syntax

SUM(Daten [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, deren Summe berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabedatenbereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Anmerkung: Viele Makrofunktionen setzen die Schlüsselwortparameter {ALL | COL | ROW} ein. Diese Schlüsselwörter gelten nicht für **IBM Kampagne**, da es sich bei den Eingabedaten immer um eine Einzelspalte oder ein Einzelfeld handelt. Das Makro verhält sich immer so, als ob das Schlüsselwort COL angegeben würde. Deshalb brauchen Sie diese Schlüsselwörter nicht anzugeben, wenn Sie **IBM Kampagne** verwenden.

Beschreibung

SUM berechnet die Summe aller Zellen im angegebenen Datenbereich. SUM gibt eine einzelne Spalte zurück.

Anmerkung: SUM ist mit der Makrofunktion TOTAL identisch.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

TEMP = SUM(3)	Erstellt eine neue Spalte TEMP, die den Wert drei enthält.
TEMP = SUM((COLUMN(3, 5, 1)))	Erstellt eine neue Spalte TEMP, die den Wert neun enthält.
TEMP = SUM(W1)	Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Summe des Inhalts von Spalte W1 darstellt.
TEMP = SUM(W1:W3)	Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Summe des Inhalts der Spalten W1, W2 und W3 darstellt.
TEMP = SUM(W1[1:5]:W4)	Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Summe der Zellen in Zeile 10-20 von Spalte W1 bis W4 darstellt.
TEMP = SUM(W1:W3, COL)	Erstellt drei neue Spalten TEMP, WX und WY. Der Einzelwert in der Spalte TEMP ist die Summe des Inhalts von Spalte W1, der Einzelwert in der Spalte WX ist die Summe des Inhalts von Spalte W2 und der Einzelwert in der Spalte WY ist die Summe des Inhalts von Spalte W3.
TEMP = SUM(W1[1:5]:W3, COL)	Erstellt drei neue Spalten TEMP, WX und WY, die jeweils einen Einzelwert enthalten. Der Wert in der Spalte TEMP ist die Summe der Zellen in Zeile 1-5 von Spalte W1, der Wert in der Spalte WX ist die Summe der Zellen in Zeile 1-5 von Spalte W2 und der Wert in der Spalte WY ist die Summe der Zellen in Zeile 1-5 von Spalte W3.
TEMP = SUM(W1:W3, ROW)	Erstellt eine neue Spalte TEMP, in der jeder Zelleneintrag die Summe der entsprechenden Zeile in Spalte W1, W2 und W3 darstellt.
TEMP = SUM(W1[1:5]:W3, ROW)	Erstellt eine neue Spalte TEMP, in der die Zellen in Zeile 1-5 die Summe der entsprechenden Zeile in Spalte W1 bis W3 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
AVG oder MEAN	Berechnet das arithmetische Mittel oder den Durchschnitt eines Zellenbereichs.
AVG_DEV	Berechnet die mittlere Abweichung eines Zellenbereichs.

TAN

Syntax

TAN(Daten [, Einheit])

Parameter

Daten

Die numerischen Werte, deren Tangens berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Einheit

Dieses optionale Schlüsselwort legt fest, ob die Eingabewerte und Ergebnisse als Grad oder als Radiant interpretiert werden. Wählen Sie eine der folgenden Angaben aus:

RADIAN - Führt die Berechnungen in Radiant aus (Standard)

DEGREE - Führt die Berechnungen in Grad aus

Wird dieser Parameter nicht angegeben, ist der Standard Radiant. (Die Umrechnung von Radiant in Grad erfolgt durch Division durch Pi und Multiplikation mit 180.)

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

TAN berechnet den Tangens der Werte im angegebenen Datenbereich. TAN gibt für jede Eingabespalte eine neue Spalte zurück, die den Tangens der Zahlen in der entsprechenden Eingabespalte enthält.

Beispiele

$TEMP = TAN(PI/4)$ oder $TEMP = TAN(PI/4, 0)$ oder $TEMP = TAN(PI/4, RADIAN)$ Erstellt eine neue Spalte TEMP, die den Wert eins enthält.
$TEMP = TAN(W1)$ Erstellt eine neue Spalte TEMP, in der jeder Wert den Tangens (in Radiant) des Inhalts von Spalte W1 darstellt.
$TEMP = TAN(W1:W3, 1)$ oder $TEMP = TAN(W1:W3, DEGREE)$ Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der Tangens des Inhalts von Spalte W1, die Werte von Spalte WX sind der Tangens des Inhalts von Spalte W2 und die Werte von Spalte WY sind der Tangens des Inhalts von Spalte W3. Alle Werte sind in Grad angegeben.
$TEMP = TAN(W1[1:5]:W2)$ Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der Tangens der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind der Tangens der entsprechenden Zeilen von Spalte W2. Alle Werte sind in Radiant angegeben.

Zugehörige Funktionen

Funktion	Beschreibung
ATAN	Berechnet den Arkustangens des Inhalts des angegebenen Datenbereichs.
COS	Berechnet den Kosinus des Inhalts des angegebenen Datenbereichs.
COT	Berechnet den Kotangens des Inhalts des angegebenen Datenbereichs.
SIN	Berechnet den Sinus des Inhalts des angegebenen Datenbereichs.
TANH	Berechnet den Hyperbeltangens des Inhalts des angegebenen Datenbereichs.

TANH

Syntax

$TANH(\text{Daten } [, \text{ Einheit}])$

Parameter

Daten

Die numerischen Werte, deren Hyperbeltangens berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Einheit

Dieses optionale Schlüsselwort legt fest, ob die Eingabewerte und Ergebnisse als Grad oder als Radiant interpretiert werden. Wählen Sie eine der folgenden Angaben aus:

RADIAN - Führt die Berechnungen in Radiant aus (Standard)

DEGREE - Führt die Berechnungen in Grad aus

Wird dieser Parameter nicht angegeben, ist der Standard Radiant. (Die Umrechnung von Radiant in Grad erfolgt durch Division durch Pi und Multiplikation mit 180.)

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Beschreibung

TANH berechnet den Hyperbeltangens der Werte im angegebenen Datenbereich. TANH gibt für jede Eingabespalte eine neue Spalte zurück, die den Hyperbeltangens der Zahlen in der entsprechenden Eingabespalte enthält. Der Hyperbeltangens einer Zahl wird wie folgt berechnet:

$$\tanh(x) = \frac{\sinh(x)}{\cosh(x)}$$

Anmerkung: Wenn der Wert x zu groß ist, wird ein Überlauffehler zurückgegeben. Dies geschieht, wenn $\tanh(x)$ den maximalen 32-Bit-Gleitkommawert überschreitet. Wenn $\cosh(x)$ null ist, gibt TANH den maximalen 32-Bit-Gleitkommawert zurück.

Beispiele

TEMP = TANH(PI) oder TEMP = TANH(PI, 0) oder TEMP = TANH(PI, RADIAN)
Erstellt eine neue Spalte TEMP, die den Wert eins enthält.
TEMP = TANH(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert den Hyperbeltangens (in Radiant) des Inhalts von Spalte W1 darstellt.
TEMP = TANH(W1:W3, 1) oder TEMP = TANH(W1:W3, DEGREE)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind jeweils der Hyperbeltangens des Inhalts von Spalte W1, die Werte von Spalte WX sind der Hyperbeltangens des Inhalts von Spalte W2 und die Werte von Spalte WY sind der Hyperbeltangens des Inhalts von Spalte W3. Alle Werte sind in Grad angegeben.
TEMP = TANH(W1[1:5]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils die Werte in Zeile 1-5 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind der Hyperbeltangens der entsprechenden Zeilen von Spalte W1, die Werte in der Spalte WX sind der Hyperbeltangens der entsprechenden Zeilen von Spalte W2. Alle Werte sind in Radiant angegeben.

Zugehörige Funktionen

Funktion	Beschreibung
ATAN	Berechnet den Arkustangens des Inhalts des angegebenen Datenbereichs.
COSH	Berechnet den Hyperbelkosinus des Inhalts des angegebenen Datenbereichs.
COT	Berechnet den Kotangens des Inhalts des angegebenen Datenbereichs.
SINH	Berechnet den Hyperbelsinus des Inhalts des angegebenen Datenbereichs.
TAN	Berechnet den Tangens des Inhalts des angegebenen Datenbereichs.

TO



Syntax

Anfang TO Ende; Anfang...Ende

Parameter

Anfang

Die erste Zahl in dem Bereich, der erstellt werden soll. Dabei kann es sich um eine ganzzahlige Konstante oder um einen Ausdruck handeln, dessen Auswertung eine ganzzahlige Konstante ergibt.

Ende

Die letzte Zahl in dem Bereich, der erstellt werden soll. Dabei kann es sich um eine ganzzahlige Konstante oder um einen Ausdruck handeln, dessen Auswertung eine ganzzahlige Konstante ergibt.

Beschreibung

TO erstellt eine einzelne Spalte, die die Ganzzahlwerte von Anfang bis Ende (einschließlich) enthält. Mithilfe dieser Makrofunktion wird die Zeitvariable in rekursiven Funktionen definiert (siehe die Makrofunktion INIT).

Anmerkung: Der Operator TO kann durch drei Punkte (...) abgekürzt werden.

Beispiele

TEMP = 1 TO 10 oder TEMP = 1...10 Erstellt eine neue Spalte TEMP, die die Werte 1-10 enthält.
TEMP = 0 TO -10 Erstellt eine neue Spalte TEMP, die die Werte 0 bis -10 enthält.

Zugehörige Funktionen

Funktion	Beschreibung
COLUMN	Erstellt neue Spalten, wobei die Eingabewerte in jeder Spalte vertikal verkettet werden.
MERGE	Erstellt eine Datengruppe durch horizontale Verkettung der Eingabewerte.

TOTAL

Syntax

TOTAL(Daten [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, deren Summe berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabedatenbereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Anmerkung: Viele Makrofunktionen setzen die Schlüsselwortparameter {ALL | COL | ROW} ein. Diese Schlüsselwörter gelten nicht für **IBM Kampagne**, da es sich bei den Eingabedaten immer um eine Einzelspalte oder ein Einzelfeld handelt. Das Makro verhält sich immer so, als ob das Schlüsselwort COL angegeben würde. Deshalb brauchen Sie diese Schlüsselwörter nicht anzugeben, wenn Sie **IBM Kampagne** verwenden.

Beschreibung

TOTAL berechnet die Summe aller Zellen im angegebenen Datenbereich.

Anmerkung: TOTAL ist mit der Makrofunktion SUM identisch.

Beispiele

TEMP = TOTAL(3)	Erstellt eine neue Spalte TEMP, die den Wert drei enthält.
TEMP = TOTAL((COLUMN(3, 5, 1)))	
	Erstellt eine neue Spalte TEMP, die den Wert neun enthält.
TEMP = TOTAL(W1)	
	Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Summe des Inhalts von Spalte W1 darstellt.
TEMP = TOTAL(W1:W3)	
	Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Summe des Inhalts der Spalten W1, W2 und W3 darstellt.
TEMP = TOTAL(W1[1:5]:W4)	
	Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Summe der Zellen in Zeile 10-20 von Spalte W1 bis W4 darstellt.
TEMP = TOTAL(W1:W3, COL)	
	Erstellt drei neue Spalten TEMP, WX und WY. Der Einzelwert in der Spalte TEMP ist die Summe des Inhalts von Spalte W1, der Einzelwert in der Spalte WX ist die Summe des Inhalts von Spalte W2 und der Einzelwert in der Spalte WY ist die Summe des Inhalts von Spalte W3.
TEMP = TOTAL(W1[1:5]:W3, COL)	
	Erstellt drei neue Spalten TEMP, WX und WY, die jeweils einen Einzelwert enthalten. Der Wert in der Spalte TEMP ist die Summe der Zellen in Zeile 1-5 von Spalte W1, der Wert in der Spalte WX ist die Summe der Zellen in Zeile 1-5 von Spalte W2 und der Wert in der Spalte WY ist die Summe der Zellen in Zeile 1-5 von Spalte W3.
TEMP = TOTAL(W1:W3, ROW)	
	Erstellt eine neue Spalte TEMP, in der jeder Zelleneintrag die Summe der entsprechenden Zeile in Spalte W1, W2 und W3 darstellt.
TEMP = TOTAL(W1[1:5]:W3, ROW)	
	Erstellt eine neue Spalte TEMP, in der die Zellen in Zeile 1-5 die Summe der entsprechenden Zeile in Spalte W1 bis W3 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
AVG oder MEAN	Berechnet das arithmetische Mittel oder den Durchschnitt eines Zellenbereichs.
AVG_DEV	Berechnet die mittlere Abweichung eines Zellenbereichs.

TRANSPOSE



Syntax

TRANSPOSE(Daten)

Parameter

Daten

Die numerischen Werte oder Zeichenfolgewerte, die umgesetzt werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

TRANSPOSE setzt den angegebenen Datenbereich um. TRANSPOSE vertauscht die horizontale und die vertikale Ausrichtung des Datenbereichs (d. h., die erste Zeile von Daten wird die erste Spalte, die zweite Zeile wird die zweite Spalte usw.).

Anmerkung: Der umgesetzte Datenbereich muss rechteckig sein. Leere Zellen in einer numerischen Spalte werden durch Nullen ersetzt. Leere Zellen in einer Zeichenfolgespalte werden durch eine leere Zeichenfolge ("") ersetzt.

Beispiele

<code>TEMP = TRANSPOSE(COLUMN(1, 2, 3))</code>
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils den Einzelwert 1, 2 bzw. 3 enthalten.
<code>TEMP = TRANSPOSE(MERGE("a", "b"))</code>
Erstellt eine neue Spalte TEMP, die a und b enthält.
<code>TEMP = TRANSPOSE(W1)</code>
Erstellt für jede Zeile in Spalte W1 eine neue Spalte. Jede Spalte enthält einen Wert, nämlich den entsprechenden Zeilenwert von Spalte W1.
<code>TEMP = TRANSPOSE(W1:W3)</code>
Erstellt für jede Zeile der längsten Spalte, W1, W2 oder W3, eine neue Spalte. Jede Spalte weist drei Zeilen auf, die die umgesetzten Werte von W1:W3 enthalten.
<code>TEMP = TRANSPOSE(W1[10:15])</code>
Erstellt sechs neue Spalten, die jeweils eine Zeile enthalten. Die erste Spalte enthält den Wert aus W1[10], die zweite Spalte enthält den Wert aus W1[11] usw.
<code>TEMP = TRANSPOSE(W1[50:99]:W2)</code>
Erstellt 100 neue Spalten. Jede Spalte weist zwei Zeilen auf, die die umgesetzten Werte der Zeilen 50-99 von Spalte W1 und W2 enthalten.

Zugehörige Funktionen

Funktion	Beschreibung
COLUMN	Erstellt neue Spalten, wobei die Eingabewerte in jeder Spalte vertikal verkettet werden.
MERGE	Erstellt eine Datengruppe durch horizontale Verkettung der Eingabewerte.

TRUNCATE

Syntax

TRUNCATE(Daten)

Parameter

Daten

Die numerischen Werte, die abgeschnitten werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

TRUNCATE berechnet den Ganzzahlanteil jedes Werts im angegebenen Datenbereich. Dabei wird für jede Eingabespalte eine neue Spalte zurückgegeben, die den ganzzahligen Anteil (den Vorkommaanteil) der Zahlen in der entsprechenden Eingabespalte enthält.

Anmerkung: Die Makrofunktionen FRACTION und TRUNCATE sind komplementär, da sie in der Summe die ursprünglichen Werte ergeben.

Beispiele

TEMP = TRUNCATE(4,3)
Erstellt eine neue Spalte TEMP, die den Wert 4 enthält.
TEMP = TRUNCATE(-2,9)
Erstellt eine neue Spalte TEMP, die den Wert -2 enthält.
TEMP = TRUNCATE(W1)
Erstellt eine neue Spalte TEMP, in der jeder Wert die Nachkommastellen des Inhalts von Spalte W1 darstellt.
TEMP = TRUNCATE(W1:W3)
Erstellt drei neue Spalten TEMP, WX und WY. Die Werte in der Spalte TEMP sind die abgeschnittenen Teile von Spalte W1, die Werte von Spalte WX sind die abgeschnittenen Teile von Spalte W2 und die Werte von Spalte WY sind die abgeschnittenen Teile von Spalte W3.
TEMP = TRUNCATE(W1[10:20])
Erstellt eine neue Spalte TEMP, in der die ersten 11 Zellen jeweils den Ganzzahlanteil der Werte in Zeile 10-20 von Spalte W1 enthalten. Die anderen Zellen in TEMP sind leer.
TEMP = TRUNCATE(W1[50:99]:W2)
Erstellt zwei neue Spalten TEMP und WX, die jeweils Werte in Zeile 1-50 enthalten (die anderen Zellen sind leer). Die Werte in der Spalte TEMP sind jeweils der Ganzzahlanteil der Zeilen von Spalte W1, die Werte in der Spalte WX sind der Ganzzahlanteil der Zeilen von Spalte W2.

Zugehörige Funktionen

Funktion	Beschreibung
CEILING	Berechnet die Obergrenze jedes Werts im angegebenen Datenbereich.
FLOOR	Berechnet die Untergrenze jedes Werts im angegebenen Datenbereich.
FRACTION	Berechnet die Nachkommastellen jedes Werts im angegebenen Datenbereich.

UPPER

Syntax

UPPER(Daten)

Parameter

Daten

Der Zeichenfolgewart, der in Großbuchstaben konvertiert werden soll.

Beschreibung

UPPER konvertiert jeden Zeichenfolgewart im angegebenen Datenbereich in Großbuchstaben. UPPER gibt für jede Eingabespalte eine neue Spalte zurück, die die in Großbuchstaben konvertierte Zeichenfolge der entsprechenden Eingabespalte enthält.

Dieses Makro ist in IBM Interact verfügbar.

Beispiele

Temp = UPPER "gold"
Erstellt eine neue Spalte TEMP, die den Wert "GOLD" enthält.
TEMP = UPPER("15. jan. 1997")
Erstellt eine neue Spalte TEMP, die die ASCII-Zeichenfolge "15. JAN. 1997" enthält.
TEMP = UPPER("Druck")
Erstellt eine neue Spalte TEMP, die die ASCII-Zeichenfolge "DRUCK" enthält.
TEMP = UPPER(W1)
Erstellt eine neue Spalte TEMP, die die Großbuchstaben jeder Zeichenfolge in der Spalte W1 enthält.

VARIANCE

Syntax

VARIANCE(Daten [, Schlüsselwort])

Parameter

Daten

Die numerischen Werte, deren Varianz berechnet werden soll. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie die Berechnung auf den Eingabedatenbereich angewendet wird. Wählen Sie eine der folgenden Angaben aus:

ALL - Wendet die Berechnung auf alle Zellen in Daten an (Standard)

COL - Führt die Berechnung für jede Spalte von Daten gesondert aus

ROW - Führt die Berechnung für jede Zeile von Daten gesondert aus

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM Kampagne finden Sie in „Formatspezifikationen“ auf Seite 9.

Weitere Informationen zur Verwendung von Schlüsselwörtern in IBM PredictiveInsight finden Sie in „Formatspezifikationen“ auf Seite 27.

Anmerkung: Viele Makrofunktionen setzen die Schlüsselwortparameter {ALL | COL | ROW} ein. Diese Schlüsselwörter gelten nicht für **IBM Kampagne**, da es sich bei den Eingabedaten immer um eine Einzelspalte oder ein Einzelfeld handelt. Das Makro verhält sich immer so, als ob das Schlüsselwort COL angegeben würde. Deshalb brauchen Sie diese Schlüsselwörter nicht anzugeben, wenn Sie **IBM Kampagne** verwenden.

Beschreibung

VARIANCE berechnet die Varianz aller Werte im angegebenen Datenbereich. Varianz ist die Standardabweichung im Quadrat. Die Varianz wird wie folgt berechnet:

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

Dabei steht x für eine Stichprobe, n für die Anzahl der Stichproben und *Mittelwert* für den Durchschnitt der Verteilung.

Anmerkung: Wenn die Anzahl der Stichproben $n = 1$ ist, gibt VARIANCE einen Fehler zurück.

Beispiele

TEMP = VARIANCE(MERGE(3, 4, 5)) oder TEMP = VARIANCE(MERGE(3, 4, 5), ALL)
Erstellt eine neue Spalte TEMP, die den Wert 0,67 enthält.
TEMP = VARIANCE(MERGE(-10, 5, 10))
Erstellt eine neue Spalte TEMP, die den Wert 72,2 enthält.
TEMP = VARIANCE(W1)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Varianz des Inhalts von Spalte W1 darstellt.
TEMP = VARIANCE(W1:W3)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Varianz des Inhalts der Spalten W1, W2 und W3 darstellt.
TEMP = VARIANCE(W1[10:20])
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Varianz der Zellen in Zeile 10-20 von Spalte W1 darstellt.
TEMP = VARIANCE(W1[1:5]:W4)
Erstellt eine neue Spalte TEMP, die einen Einzelwert enthält, der die Varianz der Zellen in Zeile 1-5 von Spalte W1 bis W4 darstellt.
TEMP = VARIANCE(W1:W3, COL)
Erstellt drei neue Spalten TEMP, WX und WY. Der Einzelwert in der Spalte TEMP ist die Varianz des Inhalts von Spalte W1, der Einzelwert in der Spalte WX ist die Varianz des Inhalts von Spalte W2 und der Einzelwert in der Spalte WY ist die Varianz des Inhalts von Spalte W3.
TEMP = VARIANCE(MERGE(1,4), COL)
Erstellt zwei neue Spalten TEMP und WX, die beide den Wert null enthalten.
TEMP = VARIANCE(W1[1:5]:W3, COL) oder TEMP = VARIANCE(W1[1:5]:W3[1:5], COL)
Erstellt drei neue Spalten TEMP, WX und WY, die jeweils einen Einzelwert enthalten. Der Wert in der Spalte TEMP ist die Varianz der Zellen in Zeile 1-5 von Spalte W1, der Wert in der Spalte WX ist die Varianz der Zellen in Zeile 1-5 von Spalte W2 und der Wert in der Spalte WY ist die Varianz der Zellen in Zeile 1-5 von Spalte W3.
TEMP = VARIANCE(W1:W3, ROW)
Erstellt eine neue Spalte TEMP, in der jeder Zelleneintrag die Varianz der entsprechenden Zeile in Spalte W1, W2 und W3 darstellt.
TEMP = VARIANCE(W1[1:5]:W3, ROW) oder TEMP = VARIANCE(W1[1:5]:W3[1:5], ROW)
Erstellt eine neue Spalte TEMP, in der die Zellen in Zeile 1-5 die Varianz der entsprechenden Zeile in Spalte W1 bis W3 enthalten. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
KURTOSIS	Berechnet die Kurtosis eines Zellenbereichs.
SKEW	Berechnet die Schiefe der Verteilung eines Zellenbereichs.

WEEKDAY

Syntax

WEEKDAY(Daten [, Schlüsselwort])

Parameter

Daten

Die ASCII-Datumsangaben, die in numerische Werte konvertiert werden sollen, die Wochentage darstellen (1-7). Dabei kann es sich um ASCII-Text in Anführungszeichen, eine Textspalte, einen Zellenbereich mit Text oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Schlüsselwort

Dieses optionale Schlüsselwort legt fest, wie Textformate für Datums- und Uhrzeitangaben interpretiert werden sollen. Wählen Sie eine der folgenden Angaben aus:

1 - mm/tt/jj (Standard)

2 - tt-*mmm*-jj

3 - mm/tt/jj hh:mm

Wird dieser Parameter nicht angegeben, ist der Standard 1.

Beschreibung

WEEKDAY konvertiert Textwerte im angegebenen Datenbereich anhand des angegebenen Umwandlungsformats für Datums- und Uhrzeitangaben in numerische Werte, die Wochentage darstellen. Die Zahl 0 steht für Sonntag, die 1 für Montag usw. bis zur 6 für Samstag. Wenn eine Zeichenfolge mit dem angegebenen Schlüsselwort nicht gepasst werden kann, gibt WEEKDAY einen Fehler zurück.

Beispiele

TEMP = WEEKDAY("1/1/95") Erstellt eine neue Spalte TEMP, die die Zahl 0 enthält (der 1. Januar 1995 ist ein Sonntag).
TEMP = WEEKDAY(W1, 2) Erstellt eine neue Spalte TEMP, die für die Zeichenfolgen in Spalte W1 Zahlen für die Wochentage enthält. Für alle Zeichenfolgen in Spalte W1 wird das Format <i>tt-<i>mmm</i>-jj</i> erwartet (andernfalls wird jeweils ??? zurückgegeben).
TEMP = WEEKDAY(W1:W3, 3) Erstellt drei neue Spalten TEMP, WX und WY. Die Spalte TEMP enthält Zahlen, die die Wochentage der Zeichenfolgen in Spalte W1 darstellen. Die Spalte WX enthält Zahlen, die die Wochentage der Zeichenfolgen in Spalte W2 darstellen. Die Spalte WY enthält Zahlen, die die Wochentage der Zeichenfolgen in Spalte W3 darstellen. Für alle Zeichenfolgen in den Spalten W1 bis W3 wird das Format <i>mm/tt/jj hh:mm</i> erwartet (andernfalls wird jeweils ??? zurückgegeben).

```
TEMP = WEEKDAY(W1[10:20]:W2, 10)
```

Erstellt zwei neue Spalten TEMP und WX. Die Spalte TEMP enthält Zahlen, die die Wochentage der Zeichenfolgen in Zeile 1-20 von Spalte W1 darstellen. Die Spalte WX enthält Zahlen, die die Wochentage der Zeichenfolgen in Zeile 1-20 von Spalte W2 darstellen. Für alle Zeichenfolgen wird das Format mm/tt/jj erwartet (andernfalls wird jeweils ??? zurückgegeben).

Zugehörige Funktionen

Funktion	Beschreibung
NUMBER	Konvertiert ASCII-Zeichenfolgen für Uhrzeit- und Datumsangaben in numerische Werte.

WEEKDAYOF

Syntax

```
WEEKDAYOF(Datumszeichenfolge [, Eingabeformat])
```

Parameter

Datumszeichenfolge

Ein Text, der ein gültiges Datum darstellt.

Eingabeformat

Eines der Schlüsselwörter in der folgenden Tabelle, das das Datumsformat von Datumszeichenfolge angibt.

Beschreibung

WEEKDAYOF gibt für das durch Datumszeichenfolge angegebene Datum den Wochentag als Zahl zwischen 0 und 6 zurück (Sonntag 0, Montag 1 usw.). Wenn Eingabeformat nicht angegeben ist, wird das Standardschlüsselwort BEGRENZ_M_T_J verwendet.

Beispiele

WEEKDAYOF("08312000", MMTTJJJJ) gibt die Zahl 4 zurück, da der Donnerstag als 4. Wochentag behandelt wird.

Anmerkung: Weitere Informationen zu gültigen Datumsformaten finden Sie unter „DATE“ auf Seite 77.

Zugehörige Funktionen

Funktion	Beschreibung
DAYOF	Gibt den Tag des Monats als Zahl zurück.
MONTHOF	Gibt den Monat des Jahres als Zahl zurück.
YEAROF	Gibt das Jahr als Zahl zurück.

XOR

Syntax

Daten1 XOR Daten2

Parameter

Daten1

Die nicht negativen Ganzzahlen, die durch bitweises exklusives Oder mit den Werten in Daten2 verknüpft werden sollen. Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Daten2

Die nicht negative(n) Ganzzahl(en), die durch bitweises exklusives Oder mit den Werten in Daten1 verknüpft werden soll(en). Dabei kann es sich um einen konstanten Wert, eine Spalte, einen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt. Die Anzahl der Spalten in Daten2 muss mit der Anzahl der Spalten in Daten1 übereinstimmen, es sei denn, bei Daten2 handelt es sich um eine Konstante. Informationen zur Formatdefinition von Daten finden Sie im Abschnitt "Makrofunktionsparameter" im Kapitel des vorliegenden Handbuchs für Ihr IBM Produkt.

Beschreibung

XOR berechnet das bitweise exklusive Oder zwischen den beiden angegebenen Datenbereichen. XOR gibt für jede Eingabespalte eine neue Spalte zurück, wobei jeweils die entsprechende Spalte von Daten1 durch bitweises exklusives Oder mit der entsprechenden Spalte von Daten2 verknüpft wird (d. h., die erste Spalte von Daten1 wird durch bitweises exklusives Oder mit der ersten Spalte von Daten2 verknüpft, die zweite Spalte mit der zweiten Spalte usw.).

Wenn es sich bei Daten2 um eine Konstante handelt, wird jeder Wert in Daten1 durch bitweises exklusives Oder mit dem betreffenden Wert verknüpft. Wenn Daten2 eine oder mehrere Spalten enthält, werden die Berechnungen zeilenweise zwischen einer Spalte von Daten1 und einer Spalte von Daten2 ausgeführt. Die erste Zeile von Daten1 wird durch bitweises exklusives Oder mit der ersten Zeile von Daten2 verknüpft, die zweite Zeile mit der zweiten Zeile usw. Diese zeilenweise ausgeführte Berechnung erzeugt für jede Zeile ein Ergebnis, bis zum letzten Wert der kürzesten Spalte.

Anmerkung: Bei dieser Makrofunktion ist die Genauigkeit auf Ganzzahlwerte kleiner als 2^{24} begrenzt. Negative Werte sind nicht zulässig.

Beispiele

```
TEMP = 3 XOR 7
```

Erstellt eine neue Spalte TEMP, die den Wert vier enthält (das bitweise exklusive Oder von 011 und 111 ergibt 100).

TEMP = W1 XOR 8
Erstellt eine neue Spalte TEMP, in der der Inhalt von Spalte W1 jeweils durch bitweises exklusives Oder mit dem Binärwert 1000 verknüpft ist.
TEMP = W1 XOR W1
Erstellt eine neue Spalte TEMP, die nur Nullen enthält (jeder Wert, der durch exklusives Oder mit sich selbst verknüpft wird, ergibt null).
TEMP = W1 XOR W2
Erstellt eine neue Spalte TEMP, in der jeder Zeilenwert von Spalte W1 durch bitweises exklusives Oder mit dem entsprechenden Zeilenwert von Spalte W2 verknüpft ist.
TEMP = W1:W3 XOR W4:W6
Erstellt drei neue Spalten TEMP, WX und WY. In der Spalte TEMP sind die Werte von Spalte W1 durch bitweises exklusives Oder mit den entsprechenden Zeilenwerten von Spalte W4 verknüpft. In der Spalte WX sind die Werte von Spalte W2 durch bitweises exklusives Oder mit den Werten von Spalte W5 verknüpft. In der Spalte WY sind die Werte von Spalte W3 durch bitweises exklusives Oder mit den Werten von Spalte W6 verknüpft.
TEMP = W1[10:20] XOR W2 oder TEMP = W1[10:20] XOR W2[1:11]
Erstellt eine neue Spalte TEMP, in deren ersten 11 Zellen die Werte in Zeile 10-20 von Spalte W1 durch bitweises exklusives Oder mit den Werten in Zeile 1-11 von Spalte W2 verknüpft sind. Die anderen Zellen in TEMP sind leer.

Zugehörige Funktionen

Funktion	Beschreibung
BIT_AND	Berechnet das bitweise Und zwischen den beiden angegebenen Datenbereichen.
BIT_NOT	Berechnet das bitweise Nicht des Inhalts des angegebenen Datenbereichs.
BIT_OR	Berechnet das bitweise Oder zwischen den beiden angegebenen Datenbereichen.

XTAB



Syntax

XTAB(Spalte1, Spalte2 [, Operator, numerische_Spalte3])

Parameter

Spalte1

Die erste Spalte, aus der die Kreuztabelle erstellt werden soll. Dabei kann es sich um eine Konstante, eine Spalte, einen einspaltigen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt.

Spalte2

Die zweite Spalte, aus der die Kreuztabelle erstellt werden soll. Dabei kann es sich um eine Konstante, eine Spalte, einen einspaltigen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt.

Operator

Eines der gültigen Operatorschlüsselwörter (siehe unten).

numerische_Spalte3

Die dritte Spalte, aus der die Kreuztabelle erstellt werden soll. Dabei kann es sich um eine Konstante, eine Spalte, einen einspaltigen Zellenbereich oder einen Ausdruck handeln, dessen Auswertung einen dieser Typen ergibt und einen numerischen Wert enthält.

Beschreibung

XTAB berechnet unterschiedliche Werte in Spalte1 und Spalte2. Anschließend wird Operator von numerische_Spalte3 am Schnittpunkt jedes Spalte1-Werts mit jedem Spalte2-Wert berechnet.

Das Operatorschlüsselwort nimmt standardmäßig den Wert COUNTOF an; in diesem Fall wird numerische_Spalte3 nicht verwendet.

Mögliche Angaben für Operator:

COUNTOF - gibt die Anzahl der Einträge an jedem Schnittpunkt zurück.

COUNTZERO - gibt die Anzahl der Einträge an jedem Schnittpunkt zurück, bei denen numerische_Spalte3 gleich 0 ist.

COUNTNONZERO - gibt die Anzahl der Einträge an jedem Schnittpunkt zurück, bei denen numerische_Spalte3 ungleich 0 ist.

COUNTNULL - gibt die Anzahl der Einträge an jedem Schnittpunkt zurück, bei denen numerische_Spalte3 den Wert NULL hat.

MINOF - gibt den kleinsten Wert von numerische_Spalte3 an jedem Schnittpunkt zurück; meldet einen fehlenden Wert, wenn es am Schnittpunkt keine Werte gibt.

MAXOF - gibt den größten Wert von numerische_Spalte3 an jedem Schnittpunkt zurück; meldet einen fehlenden Wert, wenn es am Schnittpunkt keine Werte gibt.

SUMOF - gibt die Summe aller Werte von numerische_Spalte3 an jedem Schnittpunkt zurück.

AVGOF - gibt den Durchschnitt aller von NULL verschiedenen Werte von numerische_Spalte3 an jedem Schnittpunkt zurück.

STDEVOF - gibt die Standardabweichung aller von NULL verschiedenen Werte von numerische_Spalte3 an jedem Schnittpunkt zurück.

Anmerkung: Die Makrofunktion XTAB benötigt bei umfangreichen Daten unter Umständen lange Zeit für die Berechnung. Es wird ein Fortschrittsleiste angezeigt, bis die Berechnung abgeschlossen ist. Wenn Sie die Berechnung abbrechen möch-

ten, klicken Sie auf das "X" des Fortschrittsleistes und löschen Sie die Funktionsdefinition, die die Makrofunktion XTAB enthält.

Beispiele

```
TEMP = XTAB(W1, W2)
```

Erstellt eine Folge von Zeilen und Spalten, in denen die Anzahl unterschiedlicher Werte an den Schnittpunkten der Spalten W1 und W2 berechnet wird.

```
TEMP = XTAB(W4, W5, SUMOF W6)
```

Erstellt eine Folge von Zeilen und Spalten, die die Schnittpunkte unterschiedlicher Werte der Spalten W4 und W5 darstellen. Die Kennzahl an jedem Schnittpunkt ist die Summe der Werte in Spalte W6 für die Zeilen, die dem betreffenden Schnittpunkt entsprechen.

YEAROF

Syntax

```
YEAROF(Datumszeichenfolge [, Eingabeformat])
```

Parameter

Datumszeichenfolge

Ein Text, der ein gültiges Datum darstellt.

Eingabeformat

Eines der Schlüsselwörter in der folgenden Tabelle, das das Datumsformat von Datumszeichenfolge angibt.

Beschreibung

YEAROF gibt für das durch Datumszeichenfolge angegebene Datum das Jahr als Zahl zurück. Wenn Eingabeformat nicht angegeben ist, wird das Standardschlüsselwort BEGRENZ_M_T_J verwendet.

Beispiele

```
YEAROF("31082000", TTMMJJJJ) gibt die Zahl 2000 zurück.
```

Weitere Informationen zu gültigen Datumsformaten finden Sie unter „DATE“ auf Seite 77.

Zugehörige Funktionen

Funktion	Beschreibung
DAYOF	Gibt den Tag des Monats als Zahl zurück.
MONTHOF	Gibt den Monat des Jahres als Zahl zurück.
WEEKDAYOF	Gibt den Wochentag als Zahl zurück.

Kontakt zum technischen Support von IBM

Sollte sich ein Problem nicht mithilfe der Dokumentation beheben lassen, können sich die für den Support zuständigen Kontaktpersonen Ihres Unternehmens telefonisch an den technischen Support von IBM wenden. Damit wir Ihnen möglichst schnell helfen können, beachten Sie dabei bitte die Informationen in diesem Abschnitt.

Wenn Sie wissen möchten, wer die für den Support zuständige Kontaktperson Ihres Unternehmens ist, wenden Sie sich an Ihren IBM -Administrator.

Zusammenzustellende Informationen

Halten Sie folgende Informationen bereit, wenn Sie sich an den technischen Support von IBM wenden:

- Kurze Beschreibung der Art Ihres Problems
- Detaillierte Fehlermeldungen, die beim Auftreten des Problems angezeigt werden
- Schritte zum Reproduzieren des Problems
- Entsprechende Protokolldateien, Sitzungsdateien, Konfigurationsdateien und Daten
- Informationen zu Ihrer Produkt- und Systemumgebung, die Sie entsprechend der Beschreibung unter „Systeminformationen“ abrufen können.

Systeminformationen

Bei Ihrem Anruf beim technischen Support von IBM werden Sie um verschiedene Informationen gebeten.

Sofern das Problem Sie nicht an der Anmeldung hindert, finden Sie einen Großteil der benötigten Daten auf der Info-Seite. Dort erhalten Sie Informationen zu der installierten IBM -Anwendung.

Sie können über **Hilfe > Info** (Help > About) auf die Info-Seite zugreifen. Wenn Sie nicht auf die Info-Seite zugreifen können, finden Sie die Versionsnummer der IBM -Anwendung in der Datei `version.txt` im Installationsverzeichnis jeder Anwendung.

Kontaktinformationen für den technischen Support von IBM

Falls Sie sich an den technischen Support von IBM wenden möchten, finden Sie weitere Informationen auf der Website des technischen Supports für IBM : http://www-947.ibm.com/support/entry/portal/open_service_request.

Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden.

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen zu den gegenwärtig im jeweiligen Land verfügbaren Produkten und Services erhalten Sie beim zuständigen IBM Ansprechpartner. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Services können auch andere, ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte von IBM verletzen. Die Verantwortung für den Betrieb von Produkten, Programmen und Services anderer Anbieter liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Director of Licensing IBM Europe
Middle East & Africa
Tour Descartes
2, avenue Gambetta
92066 Paris La Defense
France

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die hier enthaltenen Informationen werden in regelmäßigen Zeitabständen aktualisiert und als Neuausgabe veröffentlicht. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter werden lediglich als Service für den Kunden bereitgestellt und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängig voneinander erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

IBM Corporation
170 Tracer Lane
Waltham, MA 02451
USA

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des in diesem Dokument beschriebenen Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedingungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer kontrollierten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Gewährleistung, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können davon abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Aussagen über Pläne und Absichten von IBM unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele von IBM.

Alle von IBM angegebenen Preise sind empfohlene Richtpreise und können jederzeit ohne weitere Mitteilung geändert werden. Händlerpreise können unter Umständen von den hier genannten Preisen abweichen.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufs. Sie sollen nur die Funktionen des Lizenzprogramms illustrieren und können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Beispielanwendungsprogramme, die in Quellsprache geschrieben sind und Programmier Techniken in verschiedenen Betriebsumgebungen veranschaulichen. Sie dürfen diese Musterprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle für die Betriebsumgebung konform sind, für die diese Musterprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten. Die Beispielprogramme werden ohne Wartung (auf "as-is"-Basis) und ohne jegliche Gewährleistung zur Verfügung gestellt. IBM übernimmt keine Haftung für Schäden, die durch die Verwendung der Beispielprogramme entstehen.

Marken

IBM, das IBM Logo und [ibm.com](http://www.ibm.com) sind Marken oder eingetragene Marken der IBM Corporation. Weitere Produkt- und Servicenamen können Marken von IBM oder anderen Unternehmen sein. Eine aktuelle Liste der IBM Marken finden Sie auf der Webseite „Copyright and trademark information“ unter www.ibm.com/legal/copytrade.shtml.

Hinweise zu Datenschutzrichtlinien und Nutzungsbedingungen

IBM Software-Produkte, einschließlich Software as a Service-Lösungen, ("Softwareangebote") verwenden möglicherweise Cookies oder andere Technologien, um Informationen zur Produktverwendung zu erfassen, mit deren Hilfe die Benutzerfreundlichkeit verbessert werden kann, die Möglichkeiten der Interaktion mit dem Benutzer angepasst oder andere Zwecke verfolgt werden können. Ein Cookie ist ein Datenelement, das von einer Website an Ihren Browser gesendet wird und dann als Tag auf Ihrem Computer gespeichert werden kann, mit dem Ihr Computer identifiziert wird. In vielen Fällen werden von diesen Cookies keine personenbezogenen Daten erfasst. Wenn ein Softwareangebot, das von Ihnen verwendet wird, die Erfassung personenbezogener Daten anhand von Cookies und ähnlichen Technologien ermöglicht, werden Sie im Folgenden über die hierbei geltenden Besonderheiten informiert.

Abhängig von den implementierten Konfigurationen kann dieses Softwareangebot Sitzungscookies und permanente Cookies verwenden, mit denen der Benutzername des Benutzers und andere personenbezogene Daten zum Zwecke des Sitzungsmanagements, zur Verbesserung der Benutzerfreundlichkeit und zu anderen funktionsbezogenen Zwecken sowie zur Nutzungsüberwachung erfasst werden. Diese Cookies können deaktiviert werden. Durch die Deaktivierung kann jedoch auch die von ihnen bereitgestellte Funktionalität nicht mehr genutzt werden.

Die Erfassung personenbezogener Daten mithilfe von Cookies und ähnlichen Technologien wird durch verschiedene rechtliche Bestimmungen geregelt. Wenn die für dieses Softwareangebot implementierten Konfigurationen Ihnen als Kunde die Möglichkeit bieten, personenbezogene Daten von Endbenutzern über Cookies und andere Technologien zu erfassen, dann sollten Sie ggf. juristische Beratung zu den geltenden Gesetzen für eine solche Datenerfassung in Anspruch nehmen. Dies gilt auch in Bezug auf die Anforderungen, die vom Gesetzgeber in Bezug auf Hinweise und die Einholung von Einwilligungen vorgeschrieben werden.

IBM setzt voraus, dass Kunden folgende Bedingungen erfüllen: (1) Sie stellen einen klar erkennbaren und auffälligen Link zu den Nutzungsbedingungen der Kundenwebsite (z. B. Datenschutzerklärung) bereit. Dieser Link muss wiederum einen Link zu der Vorgehensweise von IBM und des Kunden bei der Datenerhebung und Datennutzung umfassen. (2) Sie weisen darauf hin, dass Cookies und Clear GIFs/Web-Beacons von IBM im Auftrag des Kunden auf dem Computer des Besuchers platziert werden. Dieser Hinweis muss eine Erläuterung hinsichtlich des Zwecks dieser Technologie umfassen. (3) Sie müssen in dem gesetzlich vorgeschriebenen Umfang die Einwilligung von Websitebesuchern einholen, bevor Cookies und Clear GIFs/Web-Beacons vom Kunden oder von IBM im Auftrag des Kunden auf den Geräten der Websitebesucher platziert werden. .

Weitere Informationen zur Verwendung verschiedener Technologien einschließlich der Verwendung von Cookies zu diesen Zwecken finden Sie im IBM Online Privacy Statement unter der Webadresse <http://www.ibm.com/privacy/details/us/en> im Abschnitt mit dem Titel "Cookies, Web Beacons and Other Technologies".

