

IBM Macros for IBM EMM  
Version 9.1.1  
26 novembre 2014

*Guide d'utilisation*

**IBM**

**Important**

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Remarques», à la page 165.

Certaines illustrations de ce manuel ne sont pas disponibles en français à la date d'édition.

LE PRESENT DOCUMENT EST LIVRE EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFAÇON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE.

Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. Les informations qui y sont fournies sont susceptibles d'être modifiées avant que les produits décrits ne deviennent eux-mêmes disponibles. En outre, il peut contenir des informations ou des références concernant certains produits, logiciels ou services non annoncés dans ce pays. Cela ne signifie cependant pas qu'ils y seront annoncés.

Pour plus de détails, pour toute demande d'ordre technique, ou pour obtenir des exemplaires de documents IBM, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial.

Vous pouvez également consulter les serveurs Internet suivants :

- <http://www.fr.ibm.com> (serveur IBM en France)
- <http://www.ibm.com/ca/fr> (serveur IBM au Canada)
- <http://www.ibm.com> (serveur IBM aux Etats-Unis)

*Compagnie IBM France  
Direction Qualité  
17, avenue de l'Europe  
92275 Bois-Colombes Cedex*

Cette édition s'applique à la version 9.1.0 de la famille de produits IBM Campaign et à toutes les éditions et modifications ultérieures sauf mention contraire dans les nouvelles éditions.

© Copyright IBM Corporation 1998, 2014.

---

# Table des matières

<b>Avis aux lecteurs canadiens . . . . .</b>	<b>v</b>	Configuration des dates sur votre application Web . . . . .	50
<b>Chapitre 1. Utilisation de macros dans IBM Campaign . . . . .</b>	<b>1</b>	Macro CURRENT_WEEKDAY . . . . .	51
Récapitulatifs des fonctions macro pour IBM Campaign . . . . .	1	Macro CURRENT_YEAR . . . . .	52
Fonctions statistiques . . . . .	1	DATE . . . . .	52
Fonctions mathématiques et trigonométriques . . . . .	2	Macro DATE_FORMAT . . . . .	54
Fonctions des chaînes de caractères . . . . .	6	Macro DATE_JULIAN . . . . .	55
Fonctions d'horodatage . . . . .	7	Macro DATE_STRING . . . . .	56
Fonctions de groupement . . . . .	8	Macro DAY_BETWEEN . . . . .	58
Fonctions diverses . . . . .	8	Macro DAY_FROMNOW . . . . .	59
Paramètres des fonctions macro pour IBM Campaign	8	Macro DAY_INTERVAL . . . . .	60
Spécifications de format . . . . .	9	Macro DAYOF . . . . .	60
Utilisation des constantes . . . . .	9	Macro DISTANCE . . . . .	61
<b>Chapitre 2. Macros dans IBM Interact</b>	<b>11</b>	Macro DIV . . . . .	62
Récapitulatifs des fonctions macro pour IBM Interact	11	Macro EQ . . . . .	64
Fonctions statistiques . . . . .	12	Macro EXP . . . . .	66
Fonctions mathématiques et trigonométriques . . . . .	12	Macro EXTERNALCALLOUT . . . . .	67
Fonctions des chaînes de caractères . . . . .	13	Macro FACTORIAL . . . . .	68
Fonctions d'horodatage . . . . .	14	Macro FLOOR . . . . .	69
Fonctions diverses . . . . .	15	Macro FORMAT . . . . .	70
Paramètres des fonctions macro pour IBM Interact	15	Macro FRACTION . . . . .	73
Spécifications de format . . . . .	15	Macro GE . . . . .	74
Utilisation des constantes . . . . .	16	Macro GROUPBY . . . . .	76
<b>Chapitre 3. Guide de référence des macros IBM . . . . .</b>	<b>17</b>	Macro GROUPBY_WHERE . . . . .	79
Mots clés de formats de date valides . . . . .	17	Macro GT . . . . .	79
Macro ABS . . . . .	19	Macro IF . . . . .	81
Macro ACOS . . . . .	20	Macro IN . . . . .	83
Macro ACOT . . . . .	22	Macro INT . . . . .	83
Macro ADD_MONTHS . . . . .	23	Macro INVERSE . . . . .	85
Macro AND . . . . .	24	Macro IS . . . . .	86
Macro ASIN . . . . .	26	Macro ISERROR . . . . .	87
Macro ATAN . . . . .	28	Macro ISODD . . . . .	87
Macro AVG . . . . .	29	Macro ISEVEN . . . . .	89
Macro BETWEEN . . . . .	31	Macro ISODD . . . . .	90
Macro BIT_AND . . . . .	32	Macro LE . . . . .	91
Macro BIT_NOT . . . . .	34	Macro LIKE . . . . .	92
Macro BIT_OR . . . . .	35	Macro LN ou LOG . . . . .	94
Macro BIT_XOR . . . . .	36	Macro LOG2 . . . . .	96
Macro CEILING . . . . .	38	Macro LOG10 . . . . .	97
Macro COLUMN . . . . .	39	Macro LOWER . . . . .	98
Macro COS . . . . .	41	Macro LT . . . . .	99
Macro COSH . . . . .	42	Macro LTRIM . . . . .	100
Macro COT . . . . .	44	Macro MAX . . . . .	101
Macro COUNT . . . . .	45	Macro MEAN . . . . .	103
Macro CURRENT_DATE . . . . .	46	Macro MIN . . . . .	104
Macro CURRENT_DAY . . . . .	48	Macro MINUS . . . . .	106
Macro CURRENT_JULIAN . . . . .	48	Macro MOD . . . . .	108
Macro CURRENT_MONTH . . . . .	49	Macro MONTHOF . . . . .	109
Macro CURRENT_TIME . . . . .	49	Macro MULT . . . . .	110
		Macro NE . . . . .	112
		Macro NOT . . . . .	114
		Macro NUMBER . . . . .	115
		Macro OR . . . . .	121
		Macro POSITION . . . . .	123
		Macro PLUS . . . . .	125
		Macro POW . . . . .	126

Macro RANDOM . . . . .	128
Macro RANDOM_GAUSS . . . . .	129
Macro ROUND. . . . .	130
Macro ROWNUM. . . . .	132
Macro RTRIM . . . . .	132
Macro SIGN. . . . .	132
Macro SIN . . . . .	133
Macro SINH. . . . .	135
Macro SQRT. . . . .	136
Macro STDV ou STDEV. . . . .	138
Macro STRING_CONCAT . . . . .	140
Macro STRING_HEAD . . . . .	141
Macro STRING_LENGTH . . . . .	142
Macro STRING_PROPER . . . . .	143
Macro STRING_SEG . . . . .	144
Macro STRING_TAIL. . . . .	145
Macro SUBSTR ou SUBSTRING . . . . .	147
Macro SUM . . . . .	148

Macro TAN . . . . .	149
Macro TANH . . . . .	151
Macro TOTAL . . . . .	152
Macro TRUNCATE . . . . .	154
Macro UPPER . . . . .	155
Macro VARIANCE . . . . .	156
Macro WEEKDAY . . . . .	158
Macro WEEKDAYOF . . . . .	159
Macro XOR . . . . .	160
Macro YEAROF . . . . .	161

**Comment contacter le support technique IBM . . . . . 163**

<b>Remarques . . . . . 165</b>
Marques . . . . . 167
Règles de confidentialité et conditions d'utilisation 167

---

## Avis aux lecteurs canadiens

Le présent document a été traduit en France. Voici les principales différences et particularités dont vous devez tenir compte.

### Illustrations

Les illustrations sont fournies à titre d'exemple. Certaines peuvent contenir des données propres à la France.

### Terminologie

La terminologie des titres IBM peut différer d'un pays à l'autre. Reportez-vous au tableau ci-dessous, au besoin.

IBM France	IBM Canada
ingénieur commercial	représentant
agence commerciale	succursale
ingénieur technico-commercial	informaticien
inspecteur	technicien du matériel

### Claviers

Les lettres sont disposées différemment : le clavier français est de type AZERTY, et le clavier français-canadien de type QWERTY.

### OS/2 et Windows - Paramètres canadiens

Au Canada, on utilise :

- les pages de codes 850 (multilingue) et 863 (français-canadien),
- le code pays 002,
- le code clavier CF.

### Nomenclature

Les touches présentées dans le tableau d'équivalence suivant sont libellées différemment selon qu'il s'agit du clavier de la France, du clavier du Canada ou du clavier des États-Unis. Reportez-vous à ce tableau pour faire correspondre les touches françaises figurant dans le présent document aux touches de votre clavier.

France	Canada	Etats-Unis
 (Pos1)		Home
Fin	Fin	End
 (PgAr)		PgUp
 (PgAv)		PgDn
Inser	Inser	Ins
Suppr	Suppr	Del
Echap	Echap	Esc
Attn	Intrp	Break
Impr écran	ImpEc	PrtSc
Verr num	Num	Num Lock
Arrêt défil	Défil	Scroll Lock
 (Verr maj)	FixMaj	Caps Lock
AltGr	AltCar	Alt (à droite)

## Brevets

Il est possible qu'IBM détienne des brevets ou qu'elle ait déposé des demandes de brevets portant sur certains sujets abordés dans ce document. Le fait qu'IBM vous fournisse le présent document ne signifie pas qu'elle vous accorde un permis d'utilisation de ces brevets. Vous pouvez envoyer, par écrit, vos demandes de renseignements relatives aux permis d'utilisation au directeur général des relations commerciales d'IBM, 3600 Steeles Avenue East, Markham, Ontario, L3R 9Z7.

## Assistance téléphonique

Si vous avez besoin d'assistance ou si vous voulez commander du matériel, des logiciels et des publications IBM, contactez IBM direct au 1 800 465-1234.

---

# Chapitre 1. Utilisation de macros dans IBM Campaign

Ce chapitre fournit des informations d'utilisation sur les macros pouvant être utilisées dans IBM® Campaign. Tous les utilisateurs IBM Campaign doivent lire ce chapitre avant de consulter le reste du présent document.

---

## Récapitulatifs des fonctions macro pour IBM Campaign

Les tableaux de cette section récapitulent les fonctions macro par catégorie.

Des pages de référence détaillées sont fournies pour chaque macro, par ordre alphabétique, dans la rubrique Chapitre 3, «Guide de référence des macros IBM», à la page 17.

**Important :** Les macros peuvent s'appliquer à la fois à IBM Campaign et IBM Interact ou à un seul de ces produits. Les descriptions des macros identifient les produits dans lesquels elles sont disponibles.

Pour des informations sur les paramètres d'entrée de la fonction de macro, voir «Paramètres des fonctions macro pour IBM Campaign», à la page 8.

**Information associée:**

«Fonctions statistiques»

«Fonctions mathématiques et trigonométriques», à la page 2

«Fonctions des chaînes de caractères», à la page 6

«Fonctions d'horodatage», à la page 7

«Fonctions de groupement», à la page 8

«Fonctions diverses», à la page 8

## Fonctions statistiques

Nom de la macro	Renvoie	Description
AVG	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la moyenne arithmétique d'une plage de cibles
COUNT	Valeur unique dans une nouvelle colonne.	Compte le nombre de valeurs d'une plage de données spécifiée

Nom de la macro	Renvoie	Description
MAX	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la valeur maximale d'une plage de cibles
MEAN	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la moyenne arithmétique d'une plage de cibles
MIN	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la valeur minimale d'une plage de cibles
STDV ou STDEV	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule l'écart type d'une plage de cibles
VARIANCE	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la variance d'une plage de cibles

## Fonctions mathématiques et trigonométriques

Nom de la macro	Renvoie	Description
ABS	Une colonne par colonne d'entrée	Calcule la valeur absolue du contenu de la plage de données spécifiée
ACOS	Une colonne par colonne d'entrée	Calcule le cosinus inverse du contenu de la plage de données spécifiée
ACOT	Une colonne par colonne d'entrée	Calcule l'arc cotangente du contenu de la plage de données spécifiée



Nom de la macro	Renvoi	Description
ASIN	Une colonne par colonne d'entrée	Calcule l'arc sinus du contenu de la plage de données spécifiée
ATAN	Une colonne par colonne d'entrée	Calcule l'arc tangente du contenu de la plage de données spécifiée
AVG	Une colonne par colonne d'entrée	Calcule la moyenne arithmétique des cibles de la plage de données spécifiée
BETWEEN	Une colonne par colonne d'entrée	Compare deux valeurs pour déterminer si la valeur fournie est comprise entre deux autres valeurs
CEILING	Une colonne par colonne d'entrée	Calcule le plafond de chaque valeur de la plage de données spécifiée
COLUMN	Une colonne par colonne d'entrée	Crée des colonnes, en concaténant verticalement les valeurs en entrée dans chaque colonne
COS	Une colonne par colonne d'entrée	Calcule le cosinus du contenu de la plage de données spécifiée
COSH	Une colonne par colonne d'entrée	Calcule le cosinus hyperbolique du contenu de la plage de données spécifiée
COT	Une colonne par colonne d'entrée	Calcule la cotangente du contenu de la plage de données spécifiée
COUNT	Une colonne contenant une valeur unique	Compte le nombre de cibles contenant des valeurs dans la plage de données spécifiée
EXP	Une colonne par colonne d'entrée	Calcule le nombre naturel (e) élevé à la puissance du contenu de chaque cible de la plage de données spécifiée
FACTORIAL	Une colonne par colonne d'entrée	Calcule la factorielle de chaque valeur de la plage de données spécifiée
FLOOR	Une colonne par colonne d'entrée	Calcule la partie entière de chaque valeur de la plage de données spécifiée
FRACTION	Une colonne par colonne d'entrée	Renvoie la partie décimale de chaque valeur de la plage de données spécifiée
INT	Une colonne par colonne d'entrée	Calcule la valeur entière (arrondie) du contenu de la plage de données spécifiée
INVERSE	Une colonne par colonne d'entrée	Calcule la valeur négative du contenu de la plage de données spécifiée

Nom de la macro	Renvoie	Description
LN	Une colonne par colonne d'entrée	Calcule le logarithme naturel du contenu de la plage de données spécifiée
LOG	Une colonne par colonne d'entrée	Calcule le logarithme naturel du contenu de la plage de données spécifiée
LOG2	Une colonne par colonne d'entrée	Calcule le logarithme base 2 du contenu de la plage de données spécifiée
LOG10	Une colonne par colonne d'entrée	Calcule le logarithme base 10 du contenu de la plage de données spécifiée
MAX	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la valeur maximale d'une plage de cibles
MEAN	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la moyenne arithmétique d'une plage de cibles
MIN	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la valeur minimale d'une plage de cibles
RANDOM	Une colonne avec le nombre de valeurs spécifié	Renvoie le nombre spécifié de nombres aléatoires
RANDOM_GAUSS	Une colonne avec le nombre de valeurs spécifié	Renvoie le nombre spécifié de valeurs aléatoires à partir d'une distribution de Gauss
ROUND	Une colonne par colonne d'entrée	Calcule la valeur arrondie du contenu de la plage de données spécifiée
SIGN	Une colonne par colonne d'entrée	Détermine le signe (positif ou négatif) des valeurs dans la plage de données spécifiée
SIN	Une colonne par colonne d'entrée	Calcule le sinus du contenu de la plage de données spécifiée
SINH	Une colonne par colonne d'entrée	Calcule le sinus hyperbolique du contenu de la plage de données spécifiée

Nom de la macro	Renvoie	Description
SQRT	Une colonne par colonne d'entrée	Calcule la racine carrée du contenu de la plage de données spécifiée
STDV ou STDEV	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule l'écart type d'une plage de cibles
SUM	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la somme d'une plage de cibles
TAN	Une colonne par colonne d'entrée	Calcule la tangente du contenu de la plage de données spécifiée
TANH	Une colonne par colonne d'entrée	Calcule la tangente hyperbolique du contenu de la plage de données spécifiée
TOTAL	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la somme d'une plage de cibles
TRUNCATE	Une colonne par colonne d'entrée	Renvoie la partie non décimale de chaque valeur de la plage de données spécifiée
VARIANCE	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la variance d'une plage de cibles

## Fonctions des chaînes de caractères

Nom de la macro	Renvoi	Description
FORMAT	Une colonne par colonne d'entrée	Fournit un contrôle de formatage en sortie pour les nombres et les chaînes (tel que la largeur de la sortie, l'alignement, la précision numérique, le symbole du séparateur décimal, le symbole de groupement, etc.). Renvoie la chaîne en sortie formatée.
LIKE	Une colonne par colonne d'entrée	Détermine si une chaîne de texte correspond à un modèle spécifié
LOWER	Une colonne par colonne d'entrée	Convertit une valeur de chaîne en minuscules
LTRIM	Une colonne par colonne d'entrée	Supprime les espaces de début de chaque valeur de chaîne
NUMBER	Une colonne par colonne d'entrée	Convertit les chaînes de texte ASCII des heures et des dates en valeurs numériques
POSITION	Une colonne par colonne d'entrée	Renvoie la position initiale d'un modèle dans une chaîne de texte
RTRIM	Une colonne par colonne d'entrée	Supprime les espaces de fin de chaque valeur de chaîne
STRING_CONCAT	Une colonne avec une valeur pour chaque ligne de la colonne d'entrée la plus courte	Concatène les chaînes de texte des plages de données spécifiées
STRING_HEAD	Une colonne par colonne d'entrée	Renvoie les <i>n</i> premiers caractères de chaque chaîne de la plage de données spécifiée
STRING_LENGTH	Une colonne par colonne d'entrée	Renvoie la longueur de chaque chaîne de la plage de données spécifiée
STRING_PROPER	Une colonne par colonne d'entrée	Convertit chaque valeur de chaîne en mettant en majuscule la première lettre de chaque chaîne ou toute lettre suivant un espace blanc ou un symbole (autre que le trait de soulignement) et transforme en minuscules tous les autres caractères
STRING_SEG	Une colonne par colonne d'entrée	Renvoie le segment de chaîne entre deux index spécifiés

Nom de la macro	Renvoie	Description
STRING_TAIL	Une colonne par colonne d'entrée	Renvoie les <i>n</i> derniers caractères de chaque chaîne de la plage de données spécifiée
SUBSTR ou SUBSTRING	Une colonne par colonne d'entrée	Renvoie les caractères d'une chaîne à partir d'une position initiale
UPPER	Une colonne par colonne d'entrée	Convertit une valeur de chaîne en majuscules

## Fonctions d'horodatage

Nom de la macro	Renvoie	Description
ADD_MONTHS	Une colonne par colonne d'entrée	Renvoie une date après l'ajout du nombre de mois spécifié
CURRENT_DATE	Une colonne par colonne d'entrée	Renvoie la date du jour dans le format format
CURRENT_DAY	Une colonne par colonne d'entrée	Renvoie le jour du mois en cours sous la forme d'un nombre compris entre 1 et 31
CURRENT_JULIAN	Une colonne par colonne d'entrée	Renvoie le nombre au format julien de la date du jour
CURRENT_MONTH	Une colonne par colonne d'entrée	Renvoie le mois en cours de l'année sous la forme d'un nombre compris entre 1 et 12
CURRENT_TIME	Une colonne par colonne d'entrée	Renvoie l'heure actuelle sous la forme d'une chaîne
CURRENT_WEEKDAY	Une colonne par colonne d'entrée	Renvoie le jour de la semaine en cours du mois sous la forme d'un nombre compris entre 0 et 6
CURRENT_YEAR	Une colonne par colonne d'entrée	Renvoie l'année en cours sous la forme d'un nombre
DATE	Une colonne par colonne d'entrée	Convertit une chaîne de date au format julien
DATE_FORMAT	Une colonne par colonne d'entrée	Convertit les formats de date
DATE_JULIAN	Une colonne par colonne d'entrée	Renvoie la date au format julien
DATE_STRING	Une colonne par colonne d'entrée	Renvoie la chaîne de date au format julien
DAY_BETWEEN	Une colonne par colonne d'entrée	Renvoie le nombre de jours entre deux dates
DAY_FROMNOW	Une colonne par colonne d'entrée	Renvoie le nombre de jours entre la date du jour et la date spécifiée
DAY_INTERVAL	Une colonne par colonne d'entrée	Renvoie le nombre de jours entre deux dates

Nom de la macro	Renvoie	Description
DAYOF	Une colonne par colonne d'entrée	Renvoie le jour du mois sous la forme d'un nombre
MONTHOF	Une colonne par colonne d'entrée	Renvoie le mois de l'année sous la forme d'un nombre
WEEKDAY	Une colonne par colonne d'entrée	Convertit les chaînes de date textuelles ASCII en jour de la semaine
WEEKDAYOF	Une colonne par colonne d'entrée	Renvoie le jour de la semaine de la semaine sous forme de nombre
YEAROF	Une colonne par colonne d'entrée	Renvoie l'année sous la forme d'un nombre

## Fonctions de groupement

Nom de la macro	Renvoie	Description
GROUPBY	Une colonne avec une valeur pour chaque ligne	Récapitule plusieurs lignes de données dans un groupe
GROUPBY_WHERE	Une colonne avec une valeur pour chaque ligne	Récapitule plusieurs lignes de données répondant à une condition indiquée et se trouvant dans un groupe

## Fonctions diverses

Nom de la macro	Renvoie	Description
IF	Une colonne avec une valeur pour chaque ligne de la colonne d'entrée la plus courte	Commence une instruction conditionnelle if-then-else
ISERROR	Une colonne avec une valeur pour chaque ligne de la colonne d'entrée la plus courte	Renvoie un un si une valeur de la ligne en entrée contient une cible erronée (???) ; renvoie zéro dans le cas contraire
ISEVEN	Une colonne par colonne d'entrée	Teste si les valeurs en entrée sont paires (à savoir, divisible par deux)
ISODD	Une colonne par colonne d'entrée	Teste si les valeurs en entrée sont impaires (à savoir, non divisible par deux)
ROWNUM	Une colonne par colonne d'entrée	Génère des nombres séquentiels de un à n, où n correspond au nombre d'enregistrements

---

## Paramètres des fonctions macro pour IBM Campaign

Cette section décrit les paramètres et l'utilisation des fonctions macro dans IBM Campaign.

## Spécifications de format

Cette section décrit le format de certains paramètres fréquemment utilisés. Elle s'applique à toutes les références à ces paramètres par les spécifications de fonction macro de ce chapitre.

### **data**

Le paramètre data représente une colonne de données sur laquelle une fonction macro doit agir.

Il peut s'agir d'une constante ou d'une zone. Pour plus d'informations, reportez-vous à la fonction macro spécifique.

**Remarque :** IBM Campaign ne prend pas en charge les calculs simultanés sur plusieurs zones ou sur un sous-ensemble de lignes.

Certains autres noms de paramètre utilisent également le même format que data. Les descriptions de ces paramètres font référence à cette section et ce format.

### **keyword**

Le paramètre keyword contrôle le comportement de la fonction macro. Il indique qu'un mot clé peut être spécifié (s'il est omis, la valeur par défaut est utilisée). Les options de mot clé sont répertoriées pour chaque fonction macro sous la forme suivante :

**{choice1 | choice2 | choice3}**

Sélectionnez l'option de mot clé offrant le comportement souhaité. L'option par défaut est indiquée en gras. Par exemple, pour les options suivantes :

**{RADIANs | DEGREEs}**

Les fonctions macro suivantes sont toutes deux valides :

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

**Remarque :** De nombreuses fonctions macro utilisent les paramètres de mot clé {ALL | COL | ROW}. Ces mots clés ne s'appliquent pas dans IBM Campaign car les données d'entrées correspondent toujours à une colonne ou à une zone. La macro se comporte toujours comme si le mot clé COL a été spécifié. Par conséquent, vous n'avez pas besoin de spécifier ces mots clés lorsque vous utilisez IBM Campaign.

## Utilisation des constantes

La plupart des paramètres de fonction macro acceptent des constantes numériques ou des expressions converties en constante numérique (les fonctions macro utilisant des chaînes acceptent les constantes de chaîne).

Dans les fonctions macro effectuant des opérations enregistrement par enregistrement (par exemple, l'ajout de deux colonnes numériques), l'utilisation d'une constante revient à spécifier une colonne contenant cette valeur de constante dans chaque ligne. En fait, si une constante est fournie comme paramètre d'entrée, cette constante est étendue à la même longueur que l'entrée.

Certaines fonctions macro acceptent des chaînes de texte ASCII et des constantes numériques. Les paramètres acceptant à la fois des constantes numériques et des chaînes de texte ASCII sont indiqués dans la section "Paramètres" de chaque fonction macro.

Des exemples sont fournis dans le tableau ci-après.

Définition de la fonction	Interprétation de la constante
PERCENT_UTILIZ = (CURR_BAL*100)/ CREDIT_LIM	La constante 100 est interprétée comme une colonne contenant le même nombre de lignes que la colonne CURR_BAL, chaque ligne contenant la constante 100. La zone dérivée PERCENT_UTILIZ contiendra chaque valeur de CURR_BAL multipliée par 100 et divisée par chaque valeur de CREDIT_LIM.
NAME = STRING_CONCAT ("Mr. ", LAST_NAME)	La constante "Mr. " est interprétée comme une colonne contenant le même nombre de lignes que la colonne LAST_NAME, chaque ligne contenant la constante "Mr. ". La zone dérivée NAME contiendra chacune des chaînes de texte de LAST_NAME précédée de "Mr. ".



---

## Chapitre 2. Macros dans IBM Interact

Cette section fournit des informations d'utilisation sur les macros pouvant être utilisées dans IBM Interact. Tous les utilisateurs IBM Interact doivent lire cette section avant de consulter le reste de ce guide.

### **Auxiliaire de formule et vérification de la syntaxe des expressions de macro dans IBM Interact**

**Important :** La boîte de dialogue Assistant de formule, y compris la fonction de vérification de syntaxe associée, valide actuellement les expressions de macro en fonction de ce qui est pris en charge par IBM Campaign supports. IBM Interact, toutefois, prend en charge uniquement un sous-ensemble de la fonctionnalité de macro Campaign. Vous devez donc vous assurer que les macros et les mots clés (comme les mots clés de format de date) utilisés pour Interact sont pris en charge. Recherchez les remarques relatives à Interact dans le chapitre Guide des macros de ce guide.

---

## Récapitulatifs des fonctions macro pour IBM Interact

Les tableaux dans les sections suivantes présentent en détail les macros spécifiques à IBM Interact.

Des pages de référence détaillées sont fournies pour chaque macro, par ordre alphabétique, dans la rubrique Chapitre 3, «Guide de référence des macros IBM», à la page 17.

**Important :** Les macros peuvent s'appliquer à la fois à IBM Campaign et IBM Interact ou à un seul de ces produits. Les descriptions des macros identifient les produits dans lesquels elles sont disponibles.

La rubrique «Paramètres des fonctions macro pour IBM Interact», à la page 15 fournit des informations sur les paramètres en entrée de la fonction macro pour Interact.

**Information associée:**

«Fonctions statistiques», à la page 12

«Fonctions mathématiques et trigonométriques», à la page 12

«Fonctions des chaînes de caractères», à la page 13

«Fonctions d'horodatage», à la page 14

«Fonctions diverses», à la page 15

## Fonctions statistiques

Nom de la macro	Renvoie	Description
AVG	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la moyenne arithmétique d'une plage de cibles
MAX	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la valeur maximale d'une plage de cibles
MEAN	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la moyenne arithmétique d'une plage de cibles
MIN	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la valeur minimale d'une plage de cibles
STDV ou STDEV	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule l'écart type d'une plage de cibles

## Fonctions mathématiques et trigonométriques

Nom de la macro	Renvoie	Description
AVG	Une colonne par colonne d'entrée	Calcule la moyenne arithmétique des cibles de la plage de données spécifiée

Nom de la macro	Renvoie	Description
MAX	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la valeur maximale d'une plage de cibles
MEAN	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la moyenne arithmétique d'une plage de cibles
MIN	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la valeur minimale d'une plage de cibles
STDV ou STDEV	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule l'écart type d'une plage de cibles
SUM	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la somme d'une plage de cibles
TOTAL	Valeur unique dans une nouvelle colonne pour le mot clé ALL ; une colonne avec une valeur unique par colonne d'entrée pour le mot clé COL ; une colonne avec une valeur par ligne pour le mot clé ROW.	Calcule la somme d'une plage de cibles

## Fonctions des chaînes de caractères

Nom de la macro	Renvoie	Description
LIKE	Une colonne par colonne d'entrée	Détermine si une chaîne de texte correspond à un modèle spécifié

Nom de la macro	Renvoie	Description
LOWER	Une colonne par colonne d'entrée	Convertit une valeur de chaîne en minuscules
LTRIM	Une colonne par colonne d'entrée	Supprime les espaces de début de chaque valeur de chaîne
NUMBER	Une colonne par colonne d'entrée	Convertit les chaînes de texte ASCII des heures et des dates en valeurs numériques
RTRIM	Une colonne par colonne d'entrée	Supprime les espaces de fin de chaque valeur de chaîne
STRING_CONCAT	Une colonne avec une valeur pour chaque ligne de la colonne d'entrée la plus courte	Concatène les chaînes des plages de données spécifiées
SUBSTR ou SUBSTRING	Une colonne par colonne d'entrée	Renvoie les caractères d'une chaîne à partir d'une position initiale
UPPER	Une colonne par colonne d'entrée	Convertit une valeur de chaîne en majuscules

## Fonctions d'horodatage

Nom de la macro	Renvoie	Description
ADD_MONTHS	Une colonne par colonne d'entrée	Ajoute le nombre indiqué de mois et renvoie une date
CURRENT_DATE	Une colonne par colonne d'entrée	Renvoie la date du jour dans le format format
CURRENT_DAY	Une colonne par colonne d'entrée	Renvoie le jour du mois en cours sous la forme d'un nombre compris entre 1 et 31
CURRENT_MONTH	Une colonne par colonne d'entrée	Renvoie le mois en cours de l'année sous la forme d'un nombre compris entre 1 et 12
CURRENT_WEEKDAY	Une colonne par colonne d'entrée	Renvoie le jour de la semaine en cours du mois sous la forme d'un nombre compris entre 0 et 6
CURRENT_YEAR	Une colonne par colonne d'entrée	Renvoie l'année en cours sous la forme d'un nombre
DATE	Une colonne par colonne d'entrée	Convertit une chaîne de date au format julien
DATE_FORMAT	Une colonne par colonne d'entrée	Convertit les formats de date

## Fonctions diverses

Nom de la macro	Renvoie	Description
EXTERNALCALLOUT	Valeurs telles que définies par l'application personnalisée écrite à l'aide de l'API ExternalCallout	Appelle une application personnalisée écrite à l'aide de l'API ExternalCallout.  Pour plus d'informations, reportez-vous au document <i>IBM Interact - Guide d'administration</i> .
IF	Une colonne avec une valeur pour chaque ligne de la colonne d'entrée la plus courte	Commence une instruction conditionnelle if-then-else

---

## Paramètres des fonctions macro pour IBM Interact

Cette section décrit les paramètres et l'utilisation des fonctions macro dans IBM Interact.

### Spécifications de format

Cette section décrit le format de certains paramètres fréquemment utilisés. Elle s'applique à toutes les références à ces paramètres par les spécifications de fonction macro de cette section.

#### data

Le paramètre data représente une colonne de données sur laquelle une fonction macro doit agir. Il peut s'agir d'une constante ou d'une zone. Pour plus d'informations, reportez-vous à la fonction macro spécifique.

**Remarque :** IBM Interact ne prend pas en charge les calculs simultanés sur plusieurs zones ou sur un sous-ensemble de lignes.

Certains autres noms de paramètre utilisent également le même format que data. La description de ces paramètres fait référence à cette section et ce format.

#### Mot-clé

Le paramètre keyword contrôle le comportement de la fonction macro. Il indique qu'un mot clé peut être spécifié (s'il est omis, la valeur par défaut est utilisée). Les options de mot clé sont répertoriées pour chaque fonction macro sous la forme suivante :

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

Sélectionnez l'option de mot clé offrant le comportement souhaité. L'option par défaut est indiquée en gras. Par exemple, pour les options suivantes :

```
{ RADIANS | DEGREES }
```

Les fonctions macro suivantes sont toutes deux valides :

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

**Remarque :** De nombreuses fonctions de macro utilisent les paramètres de mot clé {ALL | COL | ROW}. Ces mots clés ne s'appliquent pas dans IBM Interact car les données d'entrées correspondent toujours à une colonne ou à une zone. La macro se comporte toujours comme si le mot clé COL a été spécifié. Par conséquent, vous n'avez pas besoin de spécifier ces mots clés lorsque vous utilisez IBM Interact.

## Utilisation des constantes

La plupart des paramètres de fonction macro acceptent des constantes numériques ou des expressions converties en constante numérique (les fonctions macro utilisant des chaînes acceptent les constantes de chaîne).

Dans les fonctions macro effectuant des opérations enregistrement par enregistrement (par exemple, l'ajout de deux colonnes numériques), l'utilisation d'une constante revient à spécifier une colonne contenant cette valeur de constante dans chaque ligne. En fait, si une constante est fournie comme paramètre d'entrée, cette constante est étendue à la même longueur que l'entrée.

Certaines fonctions macro acceptent aussi bien des chaînes de texte ASCII que des constantes numériques. Les paramètres acceptant à la fois des constantes numériques et des chaînes de texte ASCII sont indiqués dans la section "Paramètres" de chaque fonction macro.

Des exemples sont fournis dans le tableau ci-après.

Définition de la fonction	Interprétation de la constante
PERCENT_UTILIZ = (CURR_BAL*100)/ CREDIT_LIM	La constante 100 est interprétée comme une colonne contenant le même nombre de lignes que la colonne CURR_BAL, chaque ligne contenant la constante 100. La zone dérivée PERCENT_UTILIZ contiendra chaque valeur de CURR_BAL multipliée par 100 et divisée par chaque valeur de CREDIT_LIM.
NAME = STRING_CONCAT ("Mr.", LAST_NAME)	La constante "Mr." est interprétée comme une colonne contenant le même nombre de lignes que la colonne LAST_NAME, chaque ligne contenant la constante "Mr.". La zone dérivée NAME contiendra chacune des chaînes de texte de LAST_NAME précédée de "Mr."

**Remarque :** Les constantes, telles que DT\_DELIM\_M\_D\_Y, requièrent des apostrophes.

---

## Chapitre 3. Guide de référence des macros IBM

Cette section fournit des informations de référence pour les fonctions macro de feuille de calcul disponibles dans la suite de plateforme marketing IBM.

Les macros disponibles sont répertoriées dans les pages qui suivent, par ordre alphabétique. Chaque macro est fournie avec sa syntaxe, les paramètres possibles et des exemples.

**Important :** N'utilisez pas des noms de fonction ou des mots clés du langage des macros IBM pour les en-têtes de colonne des tables utilisateur dans IBM Campaign, que vous effectuiez un mappage à partir d'une base de données ou d'un fichier à plat. Ces mots réservés peuvent provoquer des erreurs s'ils sont utilisés dans les en-têtes de colonne des tables mappées.

---

### Mots clés de formats de date valides

Le tableau suivant présente les mots clés pour les formats valides avec une description et un exemple de chaque.

mot clé	Description	Exemple(s)
MM	Mois sur deux chiffres	01, 02, 03, ..., 12
MMJJ	Mois sur deux chiffres et jour sur deux chiffres	31 mars correspond à 0331
MMJJAA	Mois sur deux chiffres, jour sur deux chiffres et année sur deux chiffres	31 mars 1970 correspond à 033170
MMJJAAAA	Mois sur deux chiffres, jour sur deux chiffres et année sur quatre chiffres	31 mars 1970 correspond à 03311970
DELIM_M_D	Tout mois délimité suivi du jour	31 mars, 3/31 ou 03-31
DELIM_M_D_Y	Toute combinaison de mois, jour et année délimitée	Mars 31, 1970 ou 3/31/70
DELIM_M_D_YYYY	Toute combinaison de mois, jour et année sur 4 chiffres délimitée	Mars 31, 1970 ou 3/31/1970
DELIM_Y_M	Toute année délimitée suivie du mois	Mars 70 ; 3-70 ou 3/1970
DELIM_Y_M_D	Toute combinaison d'année, de mois et de jour délimitée	1970 Mar 31 ou 70/3/31
AAMMM	Année sur 2 chiffres et mois sur 3 lettres	70MAR
AAMMMJJ	Année sur 2 chiffres, mois sur 3 lettres et jour sur 2 chiffres	70MAR31
AA	Année sur 2 chiffres	70
AAMM	Année sur 2 chiffres et mois sur 2 chiffres	7003

<b>mot clé</b>	<b>Description</b>	<b>Exemple(s)</b>
AAMMJJ	Année sur 2 chiffres, mois sur 2 chiffres et jour sur 2 chiffres	700331
AAAAMMM	Année sur 4 chiffres et mois sur 3 lettres	1970MAR
AAAAMMJJ	Année sur 4 chiffres, mois sur 3 lettres et jour sur 2 chiffres	1970MAR31
AAAA	Année sur quatre chiffres	1970
AAAAMM	Année sur 4 chiffres et mois sur 2 chiffres	197003
AAAAMDD	Année sur 4 chiffres, mois sur 2 chiffres et jour sur 2 chiffres	19700331
DELIM_M_Y	Tout mois délimité suivi de l'année	3-70, 3/70, Mar 70, Mars 1970
DELIM_D_M	Tout jour délimité suivi du mois	31-3, 31/3, 31 Mars
DELIM_D_M_Y	Toute combinaison de jour, de mois et d'année délimitée	31-MAR-70, 31/3/1970, 31 03 70
JJ	Jour sur 2 chiffres	31
JJMMM	Jour sur 2 chiffres et mois sur 3 lettres	31MAR
JJMMMAA	Jour sur 2 chiffres, mois sur 3 lettres et année sur 2 chiffres	31MAR70
JJMMMAAAA	Jour sur 2 chiffres, mois sur 3 lettres et année sur 4 chiffres	31MAR1970
JJMM	Jour sur 2 chiffres et mois sur 2 chiffres	3103
JJMMAA	Jour sur 2 chiffres, mois sur 2 chiffres et année sur 2 chiffres	310370
JJMMAAAA	Jour sur 2 chiffres, mois sur 2 chiffres et année sur 4 chiffres	31031970
MMAA	Mois sur deux chiffres et année sur deux chiffres	0370
MMAAAA	Mois sur deux chiffres et année sur 4 chiffres	031970
MMM	Mois sur 3 lettres	MAR
MMMJJ	Mois sur 3 lettres et jour sur 2 chiffres	MAR31
MMMJJAA	Mois sur 3 lettres, jour sur 2 chiffres et année sur 2 chiffres	MAR3170



mot clé	Description	Exemple(s)
MMMJJAAAA	Mois sur 3 lettres, jour sur 2 chiffres et année sur 4 chiffres	MAR311970
MMMAA	Mois sur 3 lettres et année sur 2 chiffres	MAR70
MMMAAAA	Mois sur 3 lettres et année sur 4 chiffres	MAR1970
MONTH	Mois de l'année	Janvier, Février, Mars et ainsi de suite ou Jan, Fév, Mar, etc.
WEEKDAY	Jour de la semaine	Dimanche, Lundi, Mardi, etc. (Dimanche = 0)
WKD	Jour abrégé de la semaine	Dim, Lun, Mar, etc. (Dim = 0)

## Macro ABS

La macro ABS est disponible uniquement dans IBM Campaign.

### Syntaxe

ABS(data)

### Paramètres

data

Valeurs numériques dont la valeur absolue doit être calculée. Ce paramètre peut correspondre à une valeur constante, une colonne, une plage de cibles ou une expression convertie en l'un de ces types. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

ABS calcule la valeur absolue des nombres dans la plage de données spécifiée. La valeur absolue d'un nombre correspond à sa valeur sans son signe (les nombres positifs restent inchangés et les nombres négatifs sont renvoyés sous la forme de nombres positifs). ABS renvoie une nouvelle colonne par colonne d'entrée, chacune contenant la valeur absolue des nombres de la colonne d'entrée correspondante.

### Exemples

TEMP = ABS(-3) ou TEMP = ABS(3)  Crée une colonne nommée TEMP contenant la valeur trois.
TEMP = ABS(V1)  Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond à la valeur absolue du contenu de la colonne V1.

<p>TEMP = ABS(V1:V3)</p> <p>Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux valeurs absolues du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux valeurs absolues du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux valeurs absolues du contenu de la colonne V3.</p>
<p>TEMP = ABS(V1[10:20])</p> <p>Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les valeurs absolues des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.</p>
<p>TEMP = ABS(V1[1:5]:V2)</p> <p>Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux valeurs absolues des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux valeurs absolues des lignes correspondantes de la colonne V2.</p>

## Fonctions associées

Fonction	Description
SIGN	Détermine le signe (positif ou négatif) des valeurs dans la plage de données spécifiée.

---

## Macro ACOS

est disponible uniquement dans IBM Campaign. ACOS

### Syntaxe

ACOS( data [, units\_keyword ])

### Paramètres

data

Valeurs numériques dont l'arc cosinus doit être calculé. Ce paramètre peut correspondre à une valeur constante, une colonne, une plage de cibles ou une expression convertie en l'un de ces types. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

units\_keyword

Ce mot clé facultatif détermine si les valeurs en entrée et les résultats sont interprétés comme des degrés ou des radians. Sélectionnez l'une des valeurs suivantes :

RADIAN - Effectue les calculs en radians (valeur par défaut)

DEGREE - Effectue les calculs en degrés

Si ce paramètre n'est pas spécifié, la valeur par défaut est radians. (Pour convertir des radians en degrés, divisez par PI et multipliez par 180.)

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

## Description

ACOS calcule l'arc cosinus des valeurs de la plage de données spécifiée. L'arc cosinus est l'angle dont le cosinus est le contenu de chaque cible. ACOS renvoie une nouvelle colonne par colonne d'entrée, chacune contenant l'arc cosinus des nombres de la colonne d'entrée correspondante.

Si le mot clé RADIAN est utilisé, ACOS renvoie les valeurs de la plage 0 à  $\pi$ . Si le mot clé DEGREE est utilisé, ACOS renvoie les valeurs de la plage 0 à 180.

**Remarque :** Le contenu des cibles de chaque colonne spécifiée doit avoir des valeurs comprises entre -1.0 et 1.0 inclus. Dans le cas contraire, une cible vide est renvoyée pour chaque entrée non valide.

## Exemples

TEMP = ACOS(0) ou TEMP = ACOS(0, 0) ou TEMP = ACOS(0, RADIAN)
Crée une colonne nommée TEMP contenant la valeur 1.571 ( $\pi/2$ radians).
TEMP = ACOS(0, 1) ou TEMP = ACOS(0, DEGREE)
Crée une colonne nommée TEMP contenant la valeur 90 (degrés).
TEMP = ACOS(V1)
Crée une colonne nommée TEMP, où chaque valeur correspond à l'arc cosinus (en radians) du contenu de la colonne V1.
TEMP = ACOS(V1:V3, 1)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux arc cosinus du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux arc cosinus du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux arc cosinus du contenu de la colonne V3. Toutes les valeurs sont en degrés.
TEMP = ACOS(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les arc cosinus des valeurs des lignes 10 à 20 de la colonne V1 (en radians). Les autres cibles de TEMP sont vides.
TEMP = ACOS(V1[1:5]:V2)
Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux arc cosinus des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux arc cosinus des lignes correspondantes de la colonne V2. Toutes les valeurs sont en radians.

## Fonctions associées

Fonction	Description
ACOT	Calcule l'arc cotangente du contenu de la plage de données spécifiée
ASIN	Calcule l'arc sinus du contenu de la plage de données spécifiée

Fonction	Description
ATAN	Calcule l'arc tangente du contenu de la plage de données spécifiée
COS	Calcule le cosinus du contenu de la plage de données spécifiée

---

## Macro ACOT

est disponible uniquement dans IBM Campaign. ACOT

### Syntaxe

ACOT(data [, units\_keyword])

### Paramètres

data

Valeurs numériques dont l'arc cotangente doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

units\_keyword

Ce mot clé facultatif détermine si les valeurs en entrée et les résultats sont interprétés comme des degrés ou des radians. Sélectionnez l'une des valeurs suivantes :

RADIAN - Effectue les calculs en radians (valeur par défaut)

DEGREE - Effectue les calculs en degrés

Si ce paramètre n'est pas spécifié, la valeur par défaut est radians. (Pour convertir des radians en degrés, divisez par PI et multipliez par 180.)

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

### Description

ACOT renvoie l'angle dont la cotangente correspond au contenu de chaque cible. ACOT renvoie une nouvelle colonne par colonne d'entrée, chacune contenant l'arc cotangente des nombres de la colonne d'entrée correspondante. Des nombres à virgule flottante sur 64 bits sont utilisés.

### Exemples

TEMP = ACOT(0.5), TEMP = ACOT(0.5, 0) ou TEMP = ACOT(0.5, RADIAN)
---

Crée une colonne nommée TEMP contenant la valeur 2.157 (radians).
---

<p>TEMP = ACOT(1, 1) ou TEMP = ACOT(1, DEGREE)</p> <p>Crée une colonne nommée TEMP contenant la valeur 0.022 (1/45) degrés.</p>
<p>TEMP = ACOT(0)</p> <p>Crée une colonne nommée TEMP contenant la valeur MAX32_Float en radians.</p>
<p>TEMP = ACOT(V1)</p> <p>Crée une colonne nommée TEMP, où chaque valeur correspond à l'arc cotangente (en radians) du contenu de la colonne V1.</p>
<p>TEMP = ACOT(V1:V3, 1)</p> <p>Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux arc cotangentes du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux arc cotangentes du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux arc cotangentes du contenu de la colonne V3. Toutes les valeurs sont en degrés.</p>
<p>TEMP = ACOT(V1[10:20])</p> <p>Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les arc cotangentes des valeurs des lignes 10 à 20 de la colonne V1 (en radians). Les autres cibles de TEMP sont vides.</p>
<p>TEMP = ACOT(V1[1:5]:V2)</p> <p>Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux arc cotangentes des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux arc cotangentes des lignes correspondantes de la colonne V2. Toutes les valeurs sont en radians.</p>

## Fonctions associées

Fonction	Description
ACOS	Calcule le cosinus inverse du contenu de la plage de données spécifiée
ASIN	Calcule l'arc sinus du contenu de la plage de données spécifiée
ATAN	Calcule l'arc tangente du contenu de la plage de données spécifiée
COT	Calcule la cotangente du contenu de la plage de données spécifiée

---

## Macro ADD\_MONTHS

La macro ADD\_MONTHS est disponible dans IBM Campaign.

### Syntaxe

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

## Paramètres

months

Entier représentant un nombre de mois à ajouter à date\_string.

date\_string

Chaîne de texte représentant une date valide, au format DELIM\_M\_D\_Y ou au format spécifié par l'argument input\_format facultatif.

input\_format

Format à utiliser pour la date calculée. Pour une liste des formats de date pris en charge, voir la fonction DATE\_FORMAT. Notez que input\_format détermine le format de la chaîne en entrée, mais aussi de la chaîne en sortie.

## Description

ADD\_MONTHS renvoie une date après l'ajout du nombre de mois spécifié à la chaîne date\_string spécifiée. La date est renvoyée au format par défaut (DELIM\_M\_D\_Y) ou au format spécifié par l'argument input\_format facultatif. Si vous souhaitez un format différent en sortie, utilisez DATE\_FORMAT.

Si l'augmentation du mois par le nombre de mois spécifié génère une date non valide, le résultat est calculé de sorte à correspondre au dernier jour du mois, comme illustré dans le dernier exemple ci-dessous. Si nécessaire, les années bissextiles sont prises en compte. Par exemple, l'ajout d'un mois à 31-Jan-2012 renvoie 29-Feb-2012.

## Exemples

ADD\_MONTHS(12, '06-25-11') ajoute un an (12 mois) à la date spécifiée et renvoie la date 06-25-12.

ADD\_MONTHS(3, '2011-06-25', DT\_DELIM\_Y\_M\_D) ajoute trois mois à la date spécifiée et renvoie la date 2011-09-25.

ADD\_MONTHS(1, '02-28-2011') renvoie la date 03-28-2011.

ADD\_MONTHS(1, '03-31-2012') renvoie la date 04-30-2012.

## Fonctions associées

Fonction	Description
DATE	Convertit une chaîne de date au format julien.
DATE_FORMAT	Convertit une date de input_format en output_format.

---

## Macro AND

La macro AND est disponible dans IBM Campaign et IBM Interact.

## Syntaxe

data1 AND data2 data1 && data2

## Paramètres

data1

Nombres dont le AND logique doit être calculé avec les valeurs de data2. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

data2

Nombres dont le AND logique doit être calculé avec les valeurs de data1. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans data2 doit correspondre au nombre de colonnes dans data1, à moins que data2 ne soit une constante. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

AND calcule le AND logique entre les deux plages de données spécifiées. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 dont le AND logique avec la colonne correspondante de data2 a été calculé (à savoir, le AND logique de la première colonne de data1 avec la première colonne de data, celui de la deuxième colonne avec la deuxième colonne et ainsi de suite).

Si data2 est une constante, le AND logique de chaque valeur de data1 avec cette valeur est calculé. Si data2 contient une ou plusieurs colonnes, les calculs sont effectués ligne par ligne entre une colonne de data1 et une colonne de data2. Le AND logique de la première ligne de data1 avec la première ligne de data2 est calculé, la seconde ligne avec la seconde ligne, etc. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** L'opérateur AND peut être abrégé en double perluète ( &&). Utilisez la double perluète pour séparer les deux arguments (par exemple, pour spécifier V1 AND 3, vous pouvez simplement entrer V1&&3).

## Exemples

```
TEMP = 1 AND 8 ou TEMP = 1 && 8
```

Crée une colonne nommée TEMP contenant la valeur un (tout nombre non nul prend la valeur un).

```
TEMP = V1 && 1
```

Crée une colonne nommée TEMP avec la valeur un pour chaque valeur de la colonne V1.

TEMP = V1 && V1
Crée une colonne nommée TEMP avec la valeur un pour chaque valeur non nulle de la colonne V1 et la valeur zéro pour chaque zéro de la colonne V1.
TEMP = V1 && V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 dont le AND logique est calculé avec la valeur de ligne correspondante de la colonne V2.
TEMP = V1:V3 && V4:V6
Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 dont le AND logique avec les valeurs de ligne correspondantes de la colonne V4 a été calculé. La colonne VX contient les valeurs résultant du AND logique des colonnes V2 et V5. La colonne VY contient les valeurs résultant du AND logique des colonnes V3 et V6.
TEMP = V1[10:20] && V2 ou TEMP = V1[10:20] && V2[1:11]
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent le résultat du AND logique des valeurs des lignes 10 à 20 de la colonne V1 avec les valeurs des lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
NOT	Calcule le non logique (NON) du contenu de la plage de données spécifiée
OU	Calcule le OR logique entre deux plages de données spécifiées

---

## Macro ASIN

est disponible uniquement dans IBM Campaign. ASIN

### Syntaxe

ASIN(data [, units\_keyword])

### Paramètres

data

Valeurs numériques dont l'arc sinus doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

units\_keyword

Ce mot clé facultatif détermine si les valeurs en entrée et les résultats sont interprétés comme des degrés ou des radians. Sélectionnez l'une des valeurs suivantes :

RADIAN - Effectue les calculs en radians (valeur par défaut)



DEGREE - Effectue les calculs en degrés

Si ce paramètre n'est pas spécifié, la valeur par défaut est radians. (Pour convertir des radians en degrés, divisez par PI et multipliez par 180.)

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

## Description

ASIN calcule l'arc sinus des valeurs de la plage de données spécifiée. L'arc sinus est l'angle dont le sinus est le contenu de chaque cible. ASIN renvoie une nouvelle colonne par colonne d'entrée, chacune contenant l'arc sinus des nombres de la colonne d'entrée correspondante.

Si le mot clé RADIAN est utilisé, ASIN renvoie les valeurs de la plage  $-\pi/2$  à  $\pi/2$ . Si le mot clé DEGREE est utilisé, ASIN renvoie les valeurs de la plage -90 à 90.

**Remarque :** Le contenu des cibles de chaque colonne spécifiée doit avoir des valeurs comprises entre -1.0 et 1.0 inclus. Dans le cas contraire, ??? est renvoyé pour chaque entrée non valide.

## Exemples

TEMP = ASIN(0.5), TEMP = ASIN(0.5, 0) ou TEMP = ASIN(0.5, RADIAN) Crée une colonne nommée TEMP contenant la valeur 0.524 ( $\pi/6$ radians).
TEMP = ASIN(0.5, 1) ou TEMP = ASIN(0.5, DEGREE) Crée une colonne nommée TEMP contenant la valeur 30 (degrés).
TEMP = ASIN(V1) Crée une colonne nommée TEMP, où chaque valeur correspond à l'arc sinus (en radians) du contenu de la colonne V1.
TEMP = ASIN(V1:V3, 1) Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux arc sinus du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux arc sinus du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux arc sinus du contenu de la colonne V3. Toutes les valeurs sont en degrés.
TEMP = ASIN(V1[10:20]) Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les arc sinus des valeurs des lignes 10 à 20 de la colonne V1 (en radians). Les autres cibles de TEMP sont vides.
TEMP = ASIN(V1[1:5]:V2) Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux arc sinus des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux arc sinus des lignes correspondantes de la colonne V2. Toutes les valeurs sont en radians.

## Fonctions associées

Fonction	Description
ACOS	Calcule le cosinus inverse du contenu de la plage de données spécifiée
ACOT	Calcule l'arc cotangente du contenu de la plage de données spécifiée
ATAN	Calcule l'arc tangente du contenu de la plage de données spécifiée
SIN	Calcule le sinus du contenu de la plage de données spécifiée

---

## Macro ATAN

est disponible uniquement dans IBM Campaign. ATAN

### Syntaxe

ATAN(data [, units\_keyword])

### Paramètres

data

Valeurs numériques dont l'arc tangente doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

units\_keyword

Ce mot clé facultatif détermine si les valeurs en entrée et les résultats sont interprétés comme des degrés ou des radians. Sélectionnez l'une des valeurs suivantes :

RADIAN - Effectue les calculs en radians (valeur par défaut)

DEGREE - Effectue les calculs en degrés

Si ce paramètre n'est pas spécifié, la valeur par défaut est radians. (Pour convertir des radians en degrés, divisez par PI et multipliez par 180.)

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

### Description

ATAN calcule l'arc tangente des valeurs de la plage de données spécifiée. L'arc tangente est l'angle dont la tangente est le contenu de chaque cible. ATAN renvoie une nouvelle colonne par colonne d'entrée, chacune contenant l'arc tangente des nombres de la colonne d'entrée correspondante.

Si le mot clé RADIAN est utilisé, ATAN renvoie les valeurs de la plage  $-\pi/2$  à  $\pi/2$ . Si le mot clé DEGREE est utilisé, ATAN renvoie les valeurs de la plage -90 à 90.

## Exemples

TEMP = ATAN(1), TEMP = ATAN(1, 0) ou TEMP = ATAN(1, RADIAN)
Crée une colonne nommée TEMP contenant la valeur 0.785 ( $\pi/4$ radians).
TEMP = ATAN(1, 1) ou TEMP = ATAN(1, DEGREE)
Crée une colonne nommée TEMP contenant la valeur 45 (degrés).
TEMP = ATAN(V1)
Crée une colonne nommée TEMP, où chaque valeur correspond à l'arc tangente (en radians) du contenu de la colonne V1.
TEMP = ATAN(V1:V3, 1)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux arc tangentes du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux arc tangentes du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux arc tangentes du contenu de la colonne V3. Toutes les valeurs sont en degrés.
TEMP = ATAN(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les arc tangentes des valeurs des lignes 10 à 20 de la colonne V1 (en radians). Les autres cibles de TEMP sont vides.
TEMP = ATAN(V1[1:5]:V2)
Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux arc tangentes des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux arc tangentes des lignes correspondantes de la colonne V2. Toutes les valeurs sont en radians.

## Fonctions associées

Fonction	Description
ACOS	Calcule l'arc cosinus du contenu de la plage de données spécifiée
ASIN	Calcule l'arc sinus du contenu de la plage de données spécifiée
ATAN	Calcule l'arc tangente du contenu de la plage de données spécifiée
TAN	Calcule la tangente du contenu de la plage de données spécifiée

---

## Macro AVG

est disponible dans IBM Campaign et IBM Interact. AVG

### Syntaxe

AVG(data [, keyword])

## Paramètres

data

Valeurs numériques dont la moyenne arithmétique doit être calculée. Ces valeurs peuvent être une valeur constante, une colonne, une plage de cibles ou une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

keyword

Ce mot clé facultatif détermine la manière dont le calcul est effectué sur la plage de données d'entrée. Sélectionnez l'une des mots clés suivants :

ALL - Effectue le calcul sur toutes les cibles de data (valeur par défaut)

COL - Effectue le calcul séparément pour chaque colonne de data

ROW - Effectue le calcul séparément pour chaque ligne de data

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

**Remarque :** De nombreuses fonctions de macro utilisent les paramètres de mot clé {ALL | COL | ROW}. Ces mots clés ne s'appliquent pas dans **IBMCampaign** car les données d'entrées correspondent toujours à une colonne ou à une zone. La macro se comporte toujours comme si le mot clé COL a été spécifié. Ainsi, il n'est pas nécessaire de spécifier ces mots clés lorsque vous utilisez **IBMCampaign**.

## Description

AVG calcule la moyenne arithmétique des cibles de la plage de données spécifiée. Pour calculer la moyenne arithmétique, le contenu de l'ensemble des cellules est additionné, puis le résultat est divisé par le nombre de cellules. Le nombre de colonnes renvoyé par AVG dépend de keyword.

- Si keyword a la valeur ALL, AVG renvoie une nouvelle colonne, contenant une valeur unique (la moyenne de toutes les cibles de data).
- Si keyword a la valeur COL, AVG renvoie une nouvelle colonne pour chaque colonne d'entrée. Chaque nouvelle colonne contient une valeur (moyenne de toutes les cibles de la colonne d'entrée correspondante).
- Si keyword a la valeur ROW, AVG renvoie une nouvelle colonne qui contient la moyenne de chaque ligne de data.

**Remarque :** Les cibles vides sont exclues du calcul.

**Remarque :** AVG est identique à la fonction macro MEAN.

## Exemples

```
TEMP = AVG(V1)
```

Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la moyenne arithmétique du contenu de la colonne V1.

TEMP = AVG(V1:V3)
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la moyenne arithmétique du contenu des colonnes V1, V2 et V3.
TEMP = AVG(V1[10:20])
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la moyenne arithmétique des cibles sur les lignes 10-20 de la colonne V1.
TEMP = AVG(V1[1:5]:V4)
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la moyenne arithmétique des cibles sur les lignes 1-5 des colonnes V1 à V4.
TEMP = AVG(V1:V3, COL)
Crée trois colonnes nommées TEMP, VX et VY. La valeur unique dans la colonne TEMP est la moyenne arithmétique du contenu de la colonne V1. La valeur unique dans la colonne VX est la moyenne arithmétique du contenu de la colonne V2. La valeur unique dans la colonne VY est la moyenne arithmétique du contenu de la colonne V3.
TEMP = AVG(V1[1:5]:V3, COL)
Crée trois colonnes nommées TEMP, VX et VY, contenant chacune une seule valeur. La valeur de la colonne TEMP est la moyenne arithmétique des cibles sur les lignes 1-5 de la colonne V1. La valeur de la colonne VX est la moyenne arithmétique des cibles sur les lignes 1-5 de la colonne V2. La valeur de la colonne VY est la moyenne arithmétique des cibles sur les lignes 1-5 de la colonne V3.
TEMP = AVG(V1, ROW)
Crée une colonne nommée TEMP contenant les mêmes valeurs que la colonne V1 (la moyenne arithmétique de tout nombre est lui-même).
TEMP = AVG(V1:V3, ROW)
Crée une colonne nommée TEMP dans laquelle chaque entrée de cible correspond à la moyenne arithmétique de la ligne correspondante dans les colonnes V1, V2 et V3.
TEMP = AVG(V1[1:5]:V3, ROW)
Crée une colonne nommée TEMP, dans laquelle les cibles des lignes 1 à 5 contiennent la moyenne arithmétique de la ligne correspondante dans les colonnes V1 à V3. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
SUM ou TOTAL	Calcule la somme d'une plage de cibles

---

## Macro BETWEEN

est disponible uniquement dans IBM Campaign. BETWEEN

### Syntaxe

value1 BETWEEN value2 AND value3

### Paramètres

Equivalent to value1 >= value2 AND < value3

## Description

BETWEEN est une variante spéciale du prédicat de comparaison. Les détails de ce prédicat sont importants et l'ordre des opérands a des conséquences inattendues. Voir la section des exemples.

**Remarque :** FROM et FOR utilise une syntaxe identique.

## Exemples

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

car la manière équivalente d'exprimer BETWEEN (à l'aide de AND) a un ordre spécifique qui n'a pas d'importance lorsque vous utilisez des littéraux, mais qui en a une si vous spécifiez value2 et value3 à l'aide de variables hôte, de paramètres, voire de sous-requêtes.

---

## Macro BIT\_AND

est disponible dans IBM Campaign et IBM Interact. BIT\_AND

### Syntaxe

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

### Paramètres

data1

Entiers non négatifs dont le AND bit à bit doit être calculé avec les valeurs de data2. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

data2

Entiers non négatifs dont le AND bit à bit doit être calculé avec les valeurs de data1. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans data2 doit correspondre au nombre de colonnes dans data1, à moins que data2 ne soit une constante. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

BIT calcule le AND bit à bit entre les deux plages de données spécifiées. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 dont le AND bit à bit avec la colonne correspondante de data2 a été calculé (à savoir, le AND bit à bit de la première colonne de data1 avec la première colonne de data, celui de la deuxième colonne avec la deuxième colonne et ainsi de suite).

Si data2 est une constante, le AND bit à bit de chaque valeur de data1 avec cette valeur est calculé. Si data2 contient une ou plusieurs colonnes, les calculs sont effectués ligne par ligne entre une colonne de data2 et une colonne de data2. Le

AND bit à bit de la première ligne de data1 avec la première ligne de data2 est calculé, la seconde ligne avec la seconde ligne, etc. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** La précision de cette fonction macro est limitée aux valeurs entières inférieures à  $2^{24}$ . Les valeurs négative sont interdites.

**Remarque :** L'opérateur BIT\_AND peut être abrégé en perluète (&). Utilisez le perluète pour séparer les deux arguments (par exemple, pour spécifier BIT\_AND(V1, 3), vous pouvez simplement entrer V1&3.

## Exemples

TEMP = 3 BIT_AND 7 ou TEMP = 3 & 7
Crée une colonne nommée TEMP contenant la valeur trois (le AND bit à bit de 011 et 111 est égal à 011).
TEMP = V1 & 8
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au AND bit à bit du contenu de la colonne V1 avec la valeur binaire 1000.
TEMP = V1 & V1
Crée une colonne nommée TEMP avec le même contenu que la colonne V1 (un AND entre toute valeur et elle-même génère cette même valeur).
TEMP = V1 & V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 dont le AND bit à bit est calculé avec la valeur de ligne correspondante de la colonne V2.
TEMP = V1:V3 & V4:V6
Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 dont le AND bit à bit avec les valeurs de ligne correspondantes de la colonne V4 a été calculé. La colonne VX contient les valeurs résultant du AND bit à bit des colonnes V2 et V5. La colonne VY contient les valeurs résultant du AND bit à bit des colonnes V3 et V6.
TEMP = V1[10:20] & V2 ou TEMP = V1[10:20] & V2[1:11]
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent le résultat du AND bit à bit des valeurs des lignes 10 à 20 de la colonne V1 avec les valeurs des lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
BIT_NOT	Calcule le NOT bit à bit du contenu de la plage de données spécifiée
BIT_OR	Calcule le OR bit à bit entre deux plages de données spécifiées
BIT_XOR ou XOR	Calcule le XOR bit à bit entre deux plages de données spécifiées

---

## Macro BIT\_NOT

est disponible dans IBM Campaign et IBM Interact. BIT\_NOT

### Syntaxe

BIT\_NOT data ~ data

### Paramètres

data

Entiers non négatifs dont le NOT bit à bit doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

BIT\_NOT calcule le NOT bit à bit des valeurs de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant le NOT bit à bit des valeurs des colonnes correspondantes de data.

**Remarque :** La précision de cette fonction macro est limitée aux valeurs entières inférieures à  $2^{24}$ . Les valeurs négative sont interdites.

**Remarque :** L'utilisation d'une colonne contenant le même nombre  $x$  dans chaque ligne que data revient à utiliser la constante  $x$  pour data.

**Remarque :** L'opérateur BIT\_NOT peut être abrégé en tilde (~). Utilisez le tilde avant la valeur (par exemple, pour spécifier BIT\_NOT(V1), vous pouvez simplement saisir ~V1.

### Exemples

TEMP = BIT\_NOT 3 ou TEMP = ~3

Crée une colonne nommée TEMP contenant la valeur quatre (le NOT bit à bit de 011 est égal à 100).

TEMP = ~V1

Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au NOT bit à bit du contenu de la colonne V1.

TEMP = ~V1:V3

Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent au NOT bit à bit du contenu de la colonne V1, les valeurs de la colonne VX correspondent au NOT bit à bit du contenu de la colonne V2 et les valeurs de la colonne VY correspondent au NOT bit à bit du contenu de la colonne V3.

TEMP = ~V1[100:200]

Crée une colonne nommée TEMP, dans laquelle les 101 premières cibles contiennent le NOT bit à bit des valeurs des lignes 1 à 50 de la colonne V1.



## Fonctions associées

Fonction	Description
BIT_AND	Calcule le AND bit à bit entre deux plages de données spécifiées
BIT_OR	Calcule le OR bit à bit entre deux plages de données spécifiées
BIT_XOR ou XOR	Calcule le XOR bit à bit entre deux plages de données spécifiées

---

## Macro BIT\_OR

est disponible dans IBM Campaign et IBM Interact. BIT\_OR

### Syntaxe

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

### Paramètres

data1

Entiers non négatifs dont le OR bit à bit doit être calculé avec les valeurs de data2. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

data2

Entiers non négatifs dont le OR bit à bit doit être calculé avec les valeurs de data1. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans data2 doit correspondre au nombre de colonnes dans data1, à moins que data2 ne soit une constante. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

BIT calcule le OR bit à bit entre les deux plages de données spécifiées. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 dont le OR bit à bit avec la colonne correspondante de data2 a été calculé (à savoir, le OR bit à bit de la première colonne de data1 avec la première colonne de data, celui de la deuxième colonne avec la deuxième colonne et ainsi de suite).

Si data2 est une constante, le OR bit à bit de chaque valeur de data1 avec cette valeur est calculé. Si data2 contient une ou plusieurs colonnes, les calculs sont effectués ligne par ligne entre une colonne de data2 et une colonne de data2. Le OR bit à bit de la première ligne de data1 avec la première ligne de data2 est calculé, la seconde ligne avec la seconde ligne, etc. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** La précision de cette fonction macro est limitée aux valeurs entières inférieures à  $2^{24}$ . Les valeurs négative sont interdites.

**Remarque :** L'opérateur BIT\_OR peut être abrégé en barre verticale (|). Utilisez la barre verticale pour séparer les deux colonnes (par exemple, pour spécifier BIT\_OR(V1, 3), vous pouvez simplement entrer V1|3. Vous pouvez également utiliser OR.

## Exemples

TEMP = 3 BIT_OR 7 ou TEMP = 3 OR 7 ou TEMP = 3   7
Crée une colonne nommée TEMP contenant la valeur sept (le OR bit à bit de 011 et 111 est égal à 111).
TEMP = V1   8
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au OR bit à bit du contenu de la colonne V1 avec la valeur binaire 1000.
TEMP = V1   V1
Crée une colonne nommée TEMP avec le même contenu que la colonne V1 (un OR entre toute valeur et elle-même génère cette même valeur).
TEMP = V1   V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 dont le OR bit à bit est calculé avec la valeur de ligne correspondante de la colonne V2.
TEMP = V1:V3   V4:V6
Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 dont le OR logique avec les valeurs de ligne correspondantes de la colonne V4 a été calculé. La colonne VX contient les valeurs résultant du OR logique des colonnes V2 et V5. La colonne VY contient les valeurs résultant du OR logique des colonnes V3 et V6.
TEMP = V1[10:20]   V2 ou TEMP = V1[10:20]   V2[1:11]
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent le résultat du OR bit à bit des valeurs des lignes 10 à 20 de la colonne V1 avec les valeurs des lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
BIT_AND	Calcule le AND bit à bit entre deux plages de données spécifiées
BIT_NOT	Calcule le NOT bit à bit du contenu de la plage de données spécifiée
BIT_XOR ou XOR	Calcule le XOR bit à bit entre deux plages de données spécifiées

## Macro BIT\_XOR

est disponible dans IBM Campaign et IBM Interact. BIT\_XOR

## Syntaxe

```
data1 BIT_XOR data2
```

## Paramètres

data1

Entiers non négatifs dont le XOR bit à bit doit être calculé avec les valeurs de data2. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

data2

Entiers non négatifs dont le XOR bit à bit doit être calculé avec les valeurs de data1. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans data2 doit correspondre au nombre de colonnes dans data1, à moins que data2 ne soit une constante. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

BIT\_XOR calcule le XOR bit à bit entre les deux plages de données spécifiées. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 dont le XOR bit à bit avec la colonne correspondante de data2 a été calculé (à savoir, le XOR bit à bit de la première colonne de data1 avec la première colonne de data, celui de la deuxième colonne avec la deuxième colonne et ainsi de suite).

Si data2 est une constante, le XOR bit à bit de chaque valeur de data1 avec cette valeur est calculé. Si data2 contient une ou plusieurs colonnes, les calculs sont effectués ligne par ligne entre une colonne de data1 et une colonne de data2. Le XOR bit à bit de la première ligne de data1 avec la première ligne de data2 est calculé, la seconde ligne avec la seconde ligne, etc. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** La précision de cette fonction macro est limitée aux valeurs entières inférieures à  $2^{24}$ . Les valeurs négative sont interdites.

## Exemples

```
TEMP = 3 BIT_XOR 7
```

Crée une colonne nommée TEMP contenant la valeur quatre (le XOR bit à bit de 011 et 111 est égal à 100).

```
TEMP = V1 BIT_XOR 8
```

Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au XOR bit à bit du contenu de la colonne V1 avec la valeur binaire 1000.

TEMP = V1 BIT_XOR V1
Crée une colonne nommée TEMP ne contenant que des zéros (un XOR entre toute valeur et elle-même génère zéro).
TEMP = V1 BIT_XOR V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 dont le XOR bit à bit est calculé avec la valeur de ligne correspondante de la colonne V2.
TEMP = V1:V3 BIT_XOR V4:V6
Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 dont le XOR bit à bit avec les valeurs de ligne correspondantes de la colonne V4 a été calculé. La colonne VX contient les valeurs résultant du XOR bit à bit des colonnes V2 et V5. La colonne VY contient les valeurs résultant du XOR bit à bit des colonnes V3 et V6.
TEMP = V1[10:20] BIT_XOR V2 ou TEMP = V1[10:20] BIT_XOR V2[1:11]
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent le résultat du XOR bit à bit des valeurs des lignes 10 à 20 de la colonne V1 avec les valeurs des lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
BIT_AND	Calcule le AND bit à bit entre deux plages de données spécifiées
BIT_NOT	Calcule le NOT bit à bit du contenu de la plage de données spécifiée
BIT_OR	Calcule le OR bit à bit entre deux plages de données spécifiées

---

## Macro CEILING

est disponible uniquement dans IBM Campaign. CEILING

### Syntaxe

CEILING(data)

### Paramètres

data

Valeurs numériques dont le plafond doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

CEILING calcule le plafond des valeurs de la plage de données spécifiée. Le plafond d'un nombre est l'entier le plus petit *non* inférieur à ce nombre. CEILING renvoie une nouvelle colonne par colonne d'entrée, chacune contenant le plafond des

nombres de la colonne d'entrée correspondante.

## Exemples

TEMP = CEILING(4.3) Crée une colonne nommée TEMP contenant la valeur 5.
TEMP = CEILING(2.9) Crée une colonne nommée TEMP contenant la valeur -2.
TEMP = CEILING(V1) Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au plafond du contenu de la colonne V1.
TEMP = CEILING(V1:V3) Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux plafonds du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux plafonds du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux plafonds du contenu de la colonne V3.
TEMP = CEILING(V1[10:20]) Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les plafonds des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.
TEMP = CEILING(V1[50:99]:V2) Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 50 (les autres cibles sont vides). Les valeurs de la colonne TEMP sont les plafonds des lignes de la colonne V1 et les valeurs de la colonne VX sont les plafonds des valeurs de la colonne V2.

## Fonctions associées

Fonction	Description
FLOOR ou INT	Calcule la partie entière de chaque valeur de la plage de données spécifiée
FRACTION	Renvoie la partie décimale de chaque valeur de la plage de données spécifiée
TRUNCATE	Renvoie la partie non décimale de chaque valeur de la plage de données spécifiée

---

## Macro COLUMN

est disponible uniquement dans IBM Campaign. COLUMN

### Syntaxe

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

### Paramètres

data

Valeur à utiliser lors de la création d'une colonne. Il peut s'agir d'une valeur constante (valeur numérique ou texte ASCII entre guillemets), d'une colonne, d'une

plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Ce paramètre peut être répété plusieurs fois, mais les paramètres suivants doivent avoir la même dimensionnalité (largeur de colonne) que le premier paramètre. Toutes les valeurs de tous les paramètres data doivent être des valeurs numériques ou du texte ASCII (vous ne pouvez pas mélanger des valeurs numériques et du texte). Si plusieurs paramètres data sont fournis, ils doivent tous comporter le même nombre de colonnes. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

COLUMN concatène verticalement ses entrées dans les colonnes d'un groupe de fonctions. Il renvoie le même nombre de nouvelles colonnes que celui indiqué dans chaque paramètre d'entrée. Un nombre illimité d'arguments peuvent être fournis. Tous les arguments doivent être des valeurs numériques ou des chaînes de texte ASCII (vous ne pouvez pas mélanger des valeurs numériques et du texte).

**Remarque :** La fonction macro COLUMN peut être abrégée en séparant les arguments data par des virgules, entre parenthèses (par exemple, `TEMP = MEAN((1,2,3,4), ALL)`). En dehors d'une autre fonction macro, la paire de parenthèses n'est pas nécessaire (par exemple, `V1=1,2,3` est équivalent à `V1=COLUMN(1,2,3)`).

## Exemples

<p><code>TEMP = COLUMN(3, 4, 5)</code> ou <code>TEMP = (3,4,5)</code> ou <code>TEMP = 3,4,5</code></p> <p>Crée une colonne nommée TEMP dont les trois premières cibles contiennent les valeurs 3, 4 et 5.</p>
<p><code>TEMP = COLUMN("one", "two", "three")</code></p> <p>Crée une colonne nommée TEMP dont les trois premières cibles contiennent les valeurs "one", "two" et "three".</p>
<p><code>TEMP = AVG(V1), STDV(V1)</code></p> <p>Crée une colonne nommée TEMP avec la moyenne de la colonne V1 dans la première cible et l'écart type de la colonne V1 dans la deuxième cible.</p>
<p><code>TEMP = V1:V2, V3:V4</code></p> <p>Crée deux colonnes nommées TEMP et VX où la colonne TEMP contient les valeurs de la colonne V1 suivies des valeurs de la colonne V3. La colonne VX contient les valeurs de la colonne V2 suivies des valeurs de la colonne V4.</p>
<p><code>TEMP = V1:V2, V3:V4</code></p> <p>Crée deux colonnes nommées TEMP et VX où la colonne TEMP contient les valeurs des cibles 1 à 10 de la colonne V1 suivies de toutes les valeurs de la colonne V3. La colonne VX contient les valeurs des cibles 1 à 10 de la colonne V2 suivies de toutes les valeurs de la colonne V4.</p>
<p><code>TEMP = V1:V2, V3:V4</code></p> <p>Crée deux colonnes nommées TEMP et VX, contenant chacune une valeur unique. La colonne TEMP contient la moyenne des colonnes V1 et V2. La colonne VX contient la moyenne des colonnes V3 et V4.</p>

---

## Macro COS

est disponible uniquement dans IBM Campaign. COS

### Syntaxe

`COS(data [, units_keyword])`

### Paramètres

`data`

Valeurs numériques dont le cosinus doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de `data`, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

`units_keyword`

Ce mot clé facultatif détermine si les valeurs en entrée et les résultats sont interprétés comme des degrés ou des radians. Sélectionnez l'une des valeurs suivantes :

RADIAN - Effectue les calculs en radians (valeur par défaut)

DEGREE - Effectue les calculs en degrés

Si ce paramètre n'est pas spécifié, la valeur par défaut est radians. (Pour convertir des radians en degrés, divisez par PI et multipliez par 180.)

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

### Description

COS calcule le cosinus des valeurs de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant le cosinus des nombres de la colonne d'entrée correspondante.

### Exemples

<code>TEMP = COS(PI), TEMP = COS(PI, 0) ou TEMP = COS(PI, RADIAN)</code>
Renvoie une colonne nommée TEMP contenant la valeur unique -1.
<code>TEMP = COS(90, 1) ou TEMP = COS(90, DEGREE)</code>
Renvoie une colonne nommée TEMP contenant la valeur unique zéro.
<code>TEMP = COS(V1), TEMP = COS(V1, 0) ou TEMP = COS(V1, RADIAN)</code>
Crée une colonne nommée TEMP, où chaque valeur correspond au cosinus (en radians) du contenu de la colonne V1.

TEMP = COS(V1:V3, 1)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux cosinus du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux cosinus du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux cosinus du contenu de la colonne V3. Toutes les valeurs sont en degrés.
TEMP = COS(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les cosinus des valeurs des lignes 10 à 20 de la colonne V1 (en radians). Les autres cibles de TEMP sont vides.
TEMP = COS(V1[1:5]:V2)
Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux cosinus des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux cosinus des lignes correspondantes de la colonne V2. Toutes les valeurs sont en radians.

## Fonctions associées

Fonction	Description
ACOS	Calcule l'arc cosinus du contenu de la plage de données spécifiée
COSH	Calcule le cosinus hyperbolique du contenu de la plage de données spécifiée
SIN	Calcule le sinus du contenu de la plage de données spécifiée
TAN	Calcule la tangente du contenu de la plage de données spécifiée

---

## Macro COSH

est disponible uniquement dans IBM Campaign. COSH

### Syntaxe

COSH(data [, units\_keyword])

### Paramètres

data

Valeurs numériques dont le cosinus hyperbolique doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

units\_keyword

Ce mot clé facultatif détermine si les valeurs en entrée et les résultats sont interprétés comme des degrés ou des radians. Sélectionnez l'une des valeurs suivantes :



RADIAN - Effectue les calculs en radians (valeur par défaut)

DEGREE - Effectue les calculs en degrés

Si ce paramètre n'est pas spécifié, la valeur par défaut est radians. (Pour convertir des radians en degrés, divisez par PI et multipliez par 180.)

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

## Description

COSH calcule le cosinus hyperbolique des valeurs de la plage de données spécifiée. Pour  $x$  en radians, le cosinus hyperbolique d'un nombre est :

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

où  $e$  représente le nombre naturel, 2.7182818. COSH renvoie une nouvelle colonne par colonne d'entrée, chacune contenant le cosinus hyperbolique des nombres de la colonne d'entrée correspondante.

**Remarque :** Si la valeur  $x$  est trop élevée, une erreur de dépassement est renvoyée. Cela se produit si  $\cosh(x)$  dépasse la valeur maximale à virgule flottante sur 32 bits.

## Exemples

TEMP = COSH(0), TEMP = COSH(0, 0) ou TEMP = COSH(0, RADIAN)
Renvoie une colonne nommée TEMP contenant la valeur un.
TEMP = COSH(V1)
Crée une colonne nommée TEMP, où chaque valeur correspond au cosinus hyperbolique (en radians) du contenu de la colonne V1.
TEMP = COSH(V1:V3, 1) ou TEMP = COSH(V1:V3, DEGREE)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux cosinus hyperboliques du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux cosinus hyperboliques du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux cosinus hyperboliques du contenu de la colonne V3. Toutes les valeurs sont en degrés.
TEMP = COSH(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les cosinus hyperboliques des valeurs des lignes 10 à 20 de la colonne V1 (en radians). Les autres cibles de TEMP sont vides.
TEMP = COSH(V1[1:5]:V2)
Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux cosinus hyperboliques des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux cosinus hyperboliques des lignes correspondantes de la colonne V2. Toutes les valeurs sont en radians.

## Fonctions associées

Fonction	Description
ACOS	Calcule l'arc cosinus du contenu de la plage de données spécifiée
COS	Calcule le cosinus du contenu de la plage de données spécifiée
SINH	Calcule le sinus hyperbolique du contenu de la plage de données spécifiée
TANH	Calcule la tangente hyperbolique du contenu de la plage de données spécifiée

---

## Macro COT

est disponible uniquement dans IBM Campaign. COT

### Syntaxe

COT(data [, units\_keyword])

### Paramètres

data

Valeurs numériques dont la cotangente doit être calculée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

units\_keyword

Ce mot clé facultatif détermine si les valeurs en entrée et les résultats sont interprétés comme des degrés ou des radians. Sélectionnez l'une des valeurs suivantes :

RADIAN - Effectue les calculs en radians (valeur par défaut)

DEGREE - Effectue les calculs en degrés

Si ce paramètre n'est pas spécifié, la valeur par défaut est radians. (Pour convertir des radians en degrés, divisez par PI et multipliez par 180.)

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

### Description

COT calcule la cotangente des valeurs de la plage de données spécifiée. La cotangente est la réciproque de la tangente. COT renvoie une nouvelle colonne par colonne d'entrée, chacune contenant la cotangente des nombres de la colonne d'entrée correspondante.

**Remarque :** Si une cible contient une valeur dont la tangente est zéro, alors l'arc cotangente est infini. Dans ce cas, COT renvoie le plus grand nombre à virgule flottante de 32 bits.

## Exemples

TEMP = COT(90), TEMP = COT(90, 0) ou TEMP = COT(90, RADIAN)
Renvoie une colonne nommée TEMP contenant la valeur -0.5.
TEMP = COT(0)
Renvoie une colonne nommée TEMP contenant la valeur MAX_FLOAT_32.
TEMP = COT(V1, 1) ou TEMP = COT(V1, DEGREE)
Crée une colonne nommée TEMP, où chaque valeur correspond à la cotangente du contenu (en degrés) de la colonne V1.
TEMP = COT(V1:V3, 1)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux cotangentes du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux cotangentes du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux cotangentes du contenu de la colonne V3. Toutes les valeurs sont en degrés.
TEMP = COT(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les cotangentes des valeurs des lignes 10 à 20 de la colonne V1 (en radians). Les autres cibles de TEMP sont vides.
TEMP = COT(V1[1:5]:V2)
Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux cotangentes des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux cotangentes des lignes correspondantes de la colonne V2. Toutes les valeurs sont en radians.

## Fonctions associées

Fonction	Description
ACOT	Calcule l'arc cotangente du contenu de la plage de données spécifiée
COS	Calcule le cosinus du contenu de la plage de données spécifiée
SIN	Calcule le sinus du contenu de la plage de données spécifiée
TAN	Calcule la tangente du contenu de la plage de données spécifiée

---

## Macro COUNT

est disponible uniquement dans IBM Campaign. COUNT

### Syntaxe

COUNT(data)

## Paramètres

data

Plage de cibles dont le nombre de cibles doit être compté. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

COUNT compte le nombre de valeurs dans la plage de données spécifiée. Il renvoie une nouvelle colonne contenant une valeur unique représentant le nombre de cibles contenant des valeurs à l'intérieur de la plage de données spécifiée.

**Remarque :** Le comptage d'une colonne vide renvoie la valeur zéro.

## Exemples

TEMP = COUNT(AVG(V1:V5))
Crée une colonne nommée TEMP contenant une valeur unique égale à un (la fonction AVG renvoie une seule cible dans le mode par défaut).
TEMP = COUNT(V1)
Crée une colonne nommée TEMP contenant une valeur unique indiquant le nombre de cibles contenant des valeurs dans la colonne V1.
TEMP = COUNT(V1:V3)
Crée une colonne nommée TEMP contenant une valeur unique indiquant le nombre de cibles contenant des valeurs dans les colonnes V1, V2 et V3.
TEMP = COUNT(V1[10:20])
Crée une colonne nommée TEMP contenant la valeur 11 (les plages sont inclusives), à condition que toutes les cibles contiennent des valeurs.
TEMP = COUNT(V1[1:5]:V4)
Crée une colonne nommée TEMP contenant la valeur 20 (5 cibles dans chaque colonne fois 4 colonnes = 20 cibles), à condition que toutes les cibles contiennent des valeurs.
TEMP = COUNT(V1[1:10])
Crée une colonne nommée TEMP contenant la valeur 3, à condition que les lignes 1 à 3 de la colonne V1 contiennent des valeurs et que les lignes 4 à 10 sont vides.

## Fonctions associées

Fonction	Description
SUM ou TOTAL	Calcule la somme d'une plage de cibles

---

## Macro CURRENT\_DATE

est disponible dans IBM Campaign et IBM Interact. CURRENT\_DATE

## Syntaxe

`CURRENT_DATE([format])`

## Paramètres

format

Un des mots clés du tableau ci-après spécifiant le format de date de `date_string`.

**Remarque :** Pour plus d'informations sur les formats de date disponibles, voir la section relative aux mots clés de format valide.

## Description

`CURRENT_DATE` renvoie la date du jour dans le format `format`. La date est déterminée par l'horloge sur le serveur IBM. Si aucun mot clé `format` n'est spécifié, la valeur par défaut `DELIM_M_D_Y` est utilisée.

Pour toutes les bases de données recommandées, IBM Campaign tente d'exécuter la macro `CURRENT_DATE` dans la base de données à l'aide d'un appel SQL de l'heure actuelle pris en charge par la base de données (par exemple, `SYSDATE`, `GETDATE`, `DATE` ou `TODAY`). Dans ces cas, tous les paramètres (y compris le format de la date) de cette fonction macro sont ignorés et la sortie inclut tout ce qui est renvoyé par la base de données (par exemple, un composant de temps peuvent être inclus dans la sortie). Si cela se produit et que vous ne souhaitez renvoyer que la date ou la date dans un format différent, vous pouvez écrire votre propre macro personnalisée à l'aide d'un script SQL brut ou utiliser d'autres macros IBM. Par exemple :

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

Dans certains cas, la macro `CURRENT_DATE()` est exécutée sur le serveur IBM Campaign (par exemple, lors d'une exécution sur un fichier à plat, sur une base de données non recommandée sans support SQL équivalent ou si l'expression de la macro Campaign ne peut pas être convertie dans la base de données). Dans ces cas, tous les paramètres sont reconnus et la sortie est renvoyée au format sélectionné.

**Remarque :** Tous les formats disponibles dans IBM Campaign ne sont pas pris en charge par IBM Interact.

Notez que vous devrez peut-être utiliser la macro `DATE_FORMAT` afin d'ajuster `CURRENT_DATE` pour votre type de base de données. Par exemple, la macro suivante fonctionne avec DB2 :

```
table_name = CURRENT_DATE()-1
```

Toutefois, pour Oracle, vous devez utiliser la macro `DATE_FORMAT` comme suit :

```
table_name = DATE_FORMAT(CURRENT_DATE()-1, DELIM_M_D_YYYY, '%Y-%m-%d')
```

## Exemples

Si la date du jour est le 13 septembre 2015, `CURRENT_DATE()` renvoie "09/13/15".

## Fonctions associées

Fonction	Description
DATE_FORMAT	Convertit les dates d'un format à un autre.
DATE_JULIAN	Renvoie la date au format julien de l'entrée.
DATE_STRING	Renvoie la chaîne de date au format julien.
DATE	Convertit une chaîne de date au format julien.

---

## Macro CURRENT\_DAY

est disponible dans IBM Campaign et IBM Interact. CURRENT\_DAY

### Syntaxe

CURRENT\_DAY()

### Description

CURRENT\_DAY renvoie le jour du mois en cours sous la forme d'un nombre compris entre 1 et 31. La date est déterminée par l'horloge système sur le serveur IBM.

### Exemples

Si la date du jour est le 19 juin, CURRENT\_DAY() renvoie le nombre 19.

## Fonctions associées

Fonction	Description
CURRENT_JULIAN	Renvoie le nombre au format julien de la date du jour.
CURRENT_MONTH	Renvoie le mois actuel sous la forme d'un nombre.
CURRENT_TIME	Renvoie l'heure actuelle sous la forme d'une chaîne.
CURRENT_WEEKDAY	Renvoie le jour de la semaine sous la forme d'un nombre.
CURRENT_YEAR	Renvoie l'année en cours sous la forme d'un nombre.

---

## Macro CURRENT\_JULIAN

est disponible uniquement dans IBM Campaign. CURRENT\_JULIAN

### Syntaxe

CURRENT\_JULIAN()

### Description

CURRENT\_JULIAN() renvoie le nombre julien de la date du jour (nombre de jours écoulés depuis le 1er janvier 0000). Equivalent à la macro DATE(CURRENT\_DATE()).

## Exemples

Si la date du jour est le 31 août 2000, CURRENT\_JULIAN() renvoie le nombre 730729.

## Fonctions associées

Fonction	Description
CURRENT_DAY	Renvoie le jour actuel sous la forme d'un nombre.
CURRENT_MONTH	Renvoie le mois actuel sous la forme d'un nombre.
CURRENT_TIME	Renvoie l'heure actuelle sous la forme d'une chaîne.
CURRENT_WEEKDAY	Renvoie le jour de la semaine sous la forme d'un nombre.
CURRENT_YEAR	Renvoie l'année en cours sous la forme d'un nombre.

---

## Macro CURRENT\_MONTH

est disponible dans IBM Campaign et IBM Interact. CURRENT\_MONTH

### Syntaxe

CURRENT\_MONTH()

### Description

CURRENT\_MONTH renvoie le mois en cours de l'année sous la forme d'un nombre compris entre 1 et 12.

### Exemples

Si la date du jour est le 19 juin, CURRENT\_MONTH() renvoie le nombre 6.

## Fonctions associées

Fonction	Description
CURRENT_DAY	Renvoie le jour actuel sous la forme d'un nombre.
CURRENT_JULIAN	Renvoie le nombre au format julien actuel.
CURRENT_TIME	Renvoie l'heure actuelle sous la forme d'une chaîne.
CURRENT_WEEKDAY	Renvoie le jour de la semaine sous la forme d'un nombre.
CURRENT_YEAR	Renvoie l'année en cours sous la forme d'un nombre.

---

## Macro CURRENT\_TIME

est disponible uniquement dans IBM Campaign. CURRENT\_TIME

## Syntaxe

CURRENT\_TIME()

## Description

CURRENT\_TIME renvoie l'heure actuelle sous la forme d'une chaîne. L'heure est déterminée par l'horloge système sur le serveur IBM.

## Configuration des dates sur votre application Web

Pour afficher correctement les dates sur votre application Web dans les versions actuelles d'IBM Campaign, le fichier de configuration de votre serveur dorsal doit être correctement configuré. Cela est particulièrement important pour les paramètres `dDateFormat` et `DateOutputFormatString` de la base de données qui contient les tables système. Si ces paramètres ne sont pas configurés correctement, les dates ne seront pas affichées non plus correctement dans Campaign. Vous configurez ces propriétés à l'aide de la plateforme marketing IBM.

## Définition des dates d'une langue spécifique sur votre application Web

**Remarque :** Tous les fichiers référencés sont installés par le programme d'installation de l'application Web, sauf indication contraire.

**Important :** `webappphome` fait référence au répertoire dans lequel l'application Web de Campaign a été installée. `language_code` fait référence aux paramètres de langue que vous choisissez pour votre système.

1. Editez le fichier `webappphome/conf/campaign_config.xml` pour vous assurer que `language_code` est présent dans la liste séparée par des virgules de la balise `<supportedLocales>`, comme illustré ci-dessous :  

```
<supportedLocales>en_US, language_code</supportedLocales>
```
2. Dans le répertoire `webappphome/webapp`, copiez l'intégralité de l'arborescence des répertoires `en_US` to `language_code` (la casse doit être respectée).
3. Dans `webappphome/webapp/WEB-INF/classes/resources`, copiez `StaticMessages_en_US.properties` dans `StaticMessages_ language_code.properties`. Copiez également `ErrorMessages_en_US.properties` dans `ErrorMessages_ language_code.properties`.
4. Modifiez le fichier `StaticMessages_ language_code.properties` : recherchez `DatePattern` et changez-le en `DatePattern=dd/MM/yyyy` (la casse doit être respectée).

**Remarque :** Ce format est défini par Java™. Pour des détails complets sur le format, reportez-vous à la documentation Java de `java.text.SimpleDateFormat`, à l'adresse <http://java.sun.com>. Il n'est pas nécessaire de modifier le fichier `StaticMessages.properties`.

5. Pour WebSphere : recompresser l'application Web en fichier jar.
6. Pour WebLogic : supprimez le module actuel de l'application Web.
  - a. Ajoutez le nouveau module.
  - b. Redéployez l'application Web.
  - c. Le redémarrage du programme d'écoute de Campaign n'est pas nécessaire.
7. Vérifiez que pour le paramètre de langue du navigateur Web, `language_code` est défini sur la première priorité. Pour plus de détails, voir les sections



ci-dessous : Configuration de la langue appropriée pour votre navigateur Web et Configuration de votre ordinateur pour l'affichage d'une langue spécifique.

**Remarque :** Utilisez bien un tiret et non un trait de soulignement dans `language_code`. La configuration de l'application Web est le seul endroit où un tiret est utilisé à la place d'un trait de soulignement.

8. Connectez-vous à Campaign. Dans Campaign, les dates doivent être affichées au format spécifié dans `StaticMessages_language_code.properties`.

Pour plus d'informations sur la manière de configurer l'heure pour IBM Campaign, voir la documentation *IBM Campaign*.

## Exemples

Si l'heure est 10:54 a.m., `CURRENT_TIME()` renvoie la chaîne "10:54:00 AM".

## Fonctions associées

Fonction	Description
<code>CURRENT_DAY</code>	Renvoie le jour actuel sous la forme d'un nombre.
<code>CURRENT_JULIAN</code>	Renvoie le nombre au format julien actuel.
<code>CURRENT_WEEKDAY</code>	Renvoie le jour de la semaine sous la forme d'un nombre.
<code>CURRENT_YEAR</code>	Renvoie l'année en cours sous la forme d'un nombre.

---

## Macro `CURRENT_WEEKDAY`

est disponible dans IBM Campaign et IBM Interact. `CURRENT_WEEKDAY`

### Syntaxe

`CURRENT_WEEKDAY()`

### Description

`CURRENT_WEEKDAY` renvoie le jour de la semaine en cours sous la forme d'un nombre compris entre 0 et 6. Dimanche est représenté par 0, lundi par 1, etc.

### Exemples

Si nous sommes vendredi, `CURRENT_WEEKDAY()` renvoie le nombre 5.

## Fonctions associées

Fonction	Description
<code>CURRENT_DAY</code>	Renvoie le jour actuel sous la forme d'un nombre.
<code>CURRENT_JULIAN</code>	Renvoie le nombre au format julien actuel.
<code>CURRENT_MONTH</code>	Renvoie le mois actuel sous la forme d'un nombre.

Fonction	Description
CURRENT_TIME	Renvoie l'heure actuelle sous la forme d'une chaîne.
CURRENT_YEAR	Renvoie l'année en cours sous la forme d'un nombre.

---

## Macro CURRENT\_YEAR

est disponible dans IBM Campaign et IBM Interact. CURRENT\_YEAR

### Syntaxe

CURRENT\_YEAR()

### Description

CURRENT\_YEAR renvoie l'année en cours sous la forme d'un nombre.

### Exemples

Si l'année en cours est 2000, CURRENT\_YEAR() renvoie le nombre 2000.

### Fonctions associées

Fonction	Description
CURRENT_DAY	Renvoie le jour actuel sous la forme d'un nombre.
CURRENT_JULIAN	Renvoie le nombre au format julien actuel.
CURRENT_MONTH	Renvoie le mois actuel sous la forme d'un nombre.
CURRENT_TIME	Renvoie l'heure actuelle sous la forme d'une chaîne.
CURRENT_WEEKDAY	Renvoie le jour de la semaine sous la forme d'un nombre.
MONTHOF	Renvoie le mois de l'année sous la forme d'un nombre.
WEEKDAYOF	Renvoie le jour de la semaine de la semaine sous forme de nombre.
YEAROF	Renvoie l'année sous la forme d'un nombre.

---

## DATE

### Syntaxe

DATE(input\_date, [input\_date format])

### Paramètres

date\_string

Chaîne de texte représentant une date valide.

format

Facultatif ; un des mots clés figurant dans le tableau "Mots clés des formats de date valides", qui spécifie le format de date de `date_string`.

## Description

La macro DATE d'Interact convertit une date en entrée en valeur entière de format neutre.

La macro DATE est calculée comme suit :  $DATE(X) = 365 +$  le nombre de jours entiers écoulés depuis midi le premier janvier 0001 ap. J.-C. Pour DATE (X), vous pouvez fournir un mot clé facultatif de format DATE en entrée afin de spécifier le mode d'analyse de l'élément DATE en entrée. Si aucun mot clé format n'est spécifié, la valeur par défaut DELIM\_M\_D\_Y est utilisée. Pour des informations supplémentaires sur les formats de date valides, voir Mots clés des formats de date valides.

Les formats de date sont de longueur fixe (par exemple, la date Février 28, 1970 est représentée sous la forme 02281970 au format MMDDYYYY) ou délimités (par exemple, Février 28, 1970, 2-28-1970, ou 02/28/1970 au format DELIM\_M\_D\_YY).

Dans les formats délimités, les délimiteurs sont la barre oblique (/), le tiret (-), l'espace ( ), la virgule (,) et les deux-points (:); les années peuvent être représentées sur 2 ou 4 chiffres et les mois peuvent être indiqués en toutes lettres (par exemple, Février), abrégés (par exemple, Fév) ou sous forme numérique (par exemple, 2 ou 02).

Pour toutes les années spécifiées sur deux chiffres :

- Par défaut, Interact suppose que les dates à deux chiffres délimitées sont comprises entre 1920 et 2020.
- Les années sur deux chiffres sans le millénaire (la valeur par défaut est 20, mais vous pouvez la modifier à l'aide du paramètre JVM) sont considérées comme étant dans les années 2000.
- Les années sur deux chiffres postérieures ou égales au seuil sont considérées comme étant dans les années 1900.

### Remarque :

- Les éléments DATE Formats disponibles dans Campaign ne sont pas tous pris en charge par Interact.
- Pour plus d'informations sur les années sur deux chiffres, voir Mots clés des formats de date valides.
- Pour plus d'informations sur la configuration de la suppression du millénaire pour les années sur deux chiffres, voir la section Arguments JVM dans le guide d'optimisation d'Interact.

Cette macro est disponible dans IBM Interact.

De nombreux systèmes métier utilisent le nombre de jours écoulés au format julien (depuis le premier janvier 0000). Le résultat de la macro DATE() d'Interact est lié à une date au format julien comme suit :

Format julien = DATE(...) + 1,721,059 + fraction du jour écoulée depuis midi.

Voici des valeurs DATE() utiles pour la période ap. J.-C :

- January 1, 2050 A.D renvoie 748,749.
- January 1, 2000 A.D renvoie 730,486
- January 1, 1990 A.D renvoie 726,834
- January 1, 1900 A.D renvoie 693,962
- January 1, 0001 A.D renvoie 365

**Remarque :** Conformément à la norme ISO 8601 et à la définition de schéma XML pour les objets Date et DateTime, le calendrier grégorien proleptique est utilisé pour calculer le nombre de jours écoulés. Dans ce système de gestion d'agenda, l'année hypothétique 0000 ap. J.-C est synonyme de l'année 0001 av. J.-C.

## Exemples

DATE("8/31/2000") renvoie le nombre 730,729.

DATE("8/31/2000",DELIM\_MM\_DD\_YYYY) renvoie le nombre 730,729.

DATE("2015-01-01",DELIM\_Y\_M\_D) renvoie le nombre 735,965.

DATE("01",DD), DATE("0101",MMDD) et DATE("1970-01-01",DELIM\_Y\_M\_D) renvoient le nombre 719,529.

## Fonctions associées

Fonction	Description
DATE_FORMAT	Convertit les dates d'un format à un autre.
DATE_JULIAN	Renvoie la date au format julien de l'entrée.
DATE_STRING	Renvoie la chaîne de date sous la forme d'une date au format julien.
CURRENT_DATE	Renvoie la date du jour dans un format spécifié.

---

## Macro DATE\_FORMAT

est disponible dans IBM Campaign et IBM Interact. DATE\_FORMAT

### Syntaxe

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

### Paramètres

date\_string

Texte représentant une date valide.

input\_format

Un des mots clés du tableau ci-après spécifiant le format de date de date\_string.

output\_format

Un des mots clés du tableau ci-après spécifiant le format de date en sortie souhaité.

## Description

DATE\_FORMAT() convertit une date de input\_format dans un autre format output\_format.

Si la date est de largeur fixe, elle doit être définie dans l'un des formats suivants :

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

MM correspond à un mois sur deux chiffres et MMM représente l'abréviation du mois sur trois caractères

Si la date est délimitée (tout délimiteur et notamment SPACE, DASH ou SLASH peut être utilisé), elle doit être définie dans l'un des formats suivants :

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

**Remarque :** Tous les formats disponibles dans IBM Campaign ne sont pas pris en charge par IBM Interact.

## Exemples

DATE\_FORMAT("012171", MMDDYY, MMDDYYYY) renvoie la chaîne "01211971".

**Remarque :** Pour des informations supplémentaires sur les formats de date valides, voir «DATE», à la page 52.

Fonctions associées

Fonction	Description
DATE	Convertit une chaîne de date au format julien.
DATE_JULIAN	Renvoie la date au format julien de l'entrée.
DATE_STRING	Renvoie la chaîne de date au format julien.

---

## Macro DATE\_JULIAN

est disponible uniquement dans IBM Campaign. DATE\_JULIAN

### Syntaxe

DATE\_JULIAN(year, month, day)

## Paramètres

year

Numéro d'année valide sur 2 ou 4 chiffres.

month

Numéro de mois valide compris entre 1 et 12.

day

Numéro de jour valide compris entre 1 et 31.

## Description

DATE\_JULIAN envoie la date au format julien pour l'entrée spécifiée. La date au format julien correspond au nombre de jours écoulés depuis le 1er janvier 0000.

## Exemples

DATE\_JULIAN (2000,08,31) renvoie le nombre 730729.

## Fonctions associées

Fonction	Description
DATE	Convertit une chaîne de date au format julien.
DATE_FORMAT	Convertit les dates d'un format à un autre.
DATE_STRING	Renvoie la chaîne de date au format julien.

---

## Macro DATE\_STRING

est disponible uniquement dans IBM Campaign. DATE\_STRING

### Syntaxe

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

### Paramètres

julian\_date

Nombre représentant une date au format julien ; nombre de jours écoulés depuis le 1er janvier 0000.

output\_format

Chaîne ; format de date valide.

max\_length

format\_string

## Chaîne de format incluant éventuellement

Code	Description
%a	Nom abrégé du jour de la semaine
%A	Nom complet du jour de la semaine
%b	Nom abrégé du mois
%B	Nom complet du mois
%c	Représentation de la date et de l'heure conforme aux paramètres régionaux
%d	Jour du mois (01 - 31)
%H	Heure au format 24 heures (00 - 23)
%I	Heure au format 12 heures (01 - 12)
%j	Jour de l'année (001 - 366)
%m	Mois (01 - 12)
%M	Minute (00 - 59)
%p	Indicateur AM/PM des paramètres régionaux en cours pour une horloge au format 12 heures
%S	Secondes (00 - 59)
%U	Semaine de l'année, le dimanche étant considéré comme le premier jour de la semaine (00-51)
%w	Jour de la semaine (0 - 6 ; dimanche correspond à 0)
%W	Semaine de l'année, le lundi étant considéré comme le premier jour de la semaine (00-51)
%x	Représentation de la date pour les paramètres régionaux en cours
%X	Représentation de l'heure pour les paramètres régionaux en cours
%y	Année sur 2 chiffres (00-99)
%Y	Année sur 4 chiffres. Les zéros de début dans l'année ne sont pas supprimés. Par exemple, l'année 0201 est affichée sous la forme 0201, et l'année 0001 sous la forme 0001.
%4Y	Année sur 4 chiffres. Les zéros de début dans l'année ne sont pas supprimés. Par exemple, l'année 0201 est affichée sous la forme 0201, et l'année 0001 sous la forme 0001.
%z, %Z	Nom ou abréviation du fuseau horaire ; aucune sortie si le fuseau horaire est inconnu
%%	Signe pourcentage

## Description

DATE\_STRING renvoie la chaîne de date au format julien. Si output\_format n'est pas fourni, le mot clé par défaut DELIM\_M\_D\_Y est utilisé.

## Exemples

DATE\_STRING(730729) renvoie la chaîne "08/31/00".

**Remarque :** Pour des informations supplémentaires sur les formats de date valides, voir «DATE», à la page 52.

## Fonctions associées

Fonction	Description
DATE	Convertit une chaîne de date en date au format julien.
DATE_JULIAN	Renvoie la date au format julien de l'entrée.
DATE_FORMAT	Convertit les dates d'un format à un autre.

---

## Macro DAY\_BETWEEN

est disponible uniquement dans IBM Campaign. DAY\_BETWEEN

### Syntaxe

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

### Paramètres

from\_date\_string

Texte représentant une date valide à partir de laquelle le nombre de jours écoulés doit être calculé.

to\_date\_string

Texte représentant une date valide jusqu'à laquelle le nombre de jours écoulés est calculé. Cette date doit être dans le même format que from\_date\_string.

input\_format

Un des mots clés du tableau ci-après, spécifiant le format de date de from\_date\_string et to\_date\_string.

### Description

DAY\_BETWEEN renvoie le nombre de jours compris entre from\_date\_string et to\_date\_string. Si input\_format n'est pas fourni, le mot clé par défaut DELIM\_M\_D\_Y est utilisé.

### Exemples

DAY\_BETWEEN("08/25/00", "08/31/00") renvoie le nombre 6.



**Remarque :** Pour des informations supplémentaires sur les formats de date valides, voir «DATE», à la page 52.

### Fonctions associées

Fonction	Description
DAY_FROMNOW	Renvoie le nombre de jours entre le jour actuel et une date spécifiée.
DAY_INTERVAL	Renvoie le nombre de jours entre deux dates spécifiées.

---

## Macro DAY\_FROMNOW

est disponible uniquement dans IBM Campaign. DAY\_FROMNOW

### Syntaxe

DAY\_FROMNOW(to\_year, to\_month, to\_day)

### Paramètres

to\_year

Numéro d'année valide sur 2 ou 4 chiffres.

to\_month

Numéro de mois valide compris entre 1 et 12.

to\_day

Numéro de jour valide compris entre 1 et 31.

### Description

DAY\_FROMNOW renvoie le nombre de jours entre la date actuelle et la date spécifiée par to\_year/to\_month/to\_day.

**Remarque :** Si la date spécifiée est antérieure, la valeur renvoyée est négative.

### Exemples

Si la date du jour est le 31 août 2000, DAY\_FROMNOW(2000,12,31) renvoie le nombre 122.

### Fonctions associées

Fonction	Description
DAY_BETWEEN	Renvoie le nombre de jours entre deux chaînes de date spécifiées.
DAY_INTERVAL	Renvoie le nombre de jours entre deux dates spécifiées.

---

## Macro DAY\_INTERVAL

est disponible uniquement dans IBM Campaign. DAY\_INTERVAL

### Syntaxe

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

### Paramètres

from\_year

Numéro d'année valide sur 2 ou 4 chiffres.

from\_month

Numéro de mois valide compris entre 1 et 12.

from\_day

Numéro de jour valide compris entre 1 et 31.

to\_year

Numéro d'année valide sur 2 ou 4 chiffres.

to\_month

Numéro de mois valide compris entre 1 et 12.

to\_day

Numéro de jour valide compris entre 1 et 31.

### Description

DAY\_INTERVAL renvoie le nombre de jours entre la date de début spécifiée (from\_year/from\_month/from\_day) et la date de fin spécifiée (to\_year/to\_month/to\_day).

### Exemples

DAY\_INTERVAL(2000,8,31,2000,12,31) renvoie le nombre 122.

### Fonctions associées

Fonction	Description
DAY_BETWEEN	Renvoie le nombre de jours entre deux chaînes de date spécifiées.
DAY_FROMNOW	Renvoie le nombre de jours entre le jour actuel et une date spécifiée.

---

## Macro DAYOF

est disponible uniquement dans IBM Campaign. DAYOF

## Syntaxe

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

## Paramètres

date\_string

Texte représentant une date valide.

input\_format

Un des mots clés du tableau ci-après, spécifiant le format de date de date\_string.

## Description

DAYOF renvoie le jour du mois sous la forme d'un nombre pour la date représentée par date\_string. Si input\_format n'est pas fourni, le mot clé par défaut DELIM\_M\_D\_Y est utilisé.

## Exemples

DAYOF("08/31/00") renvoie le nombre 31.

**Remarque :** Pour des informations supplémentaires sur les formats de date valides, voir «DATE», à la page 52.

---

## Macro DISTANCE

est disponible dans IBM Campaign et IBM Interact. DISTANCE

## Syntaxe

DISTANCE(lat1, long1, lat2, long2[, UNITE\_DE\_MESURE][, PRECISION])

## Paramètres

lat1

Latitude du premier point, sous forme de valeur décimale.

long1

Longitude du premier point, sous forme de valeur décimale.

lat2

Latitude du deuxième point, sous forme de valeur décimale.

long2

Longitude du deuxième point, sous forme de valeur décimale.

UNITE\_DE\_MESURE

Paramètre facultatif indiquant l'unité de mesure de la distance renvoyée. Les valeurs admises sont MILES et KILOMETERS. Si vous omettez ce paramètre, MILES est l'unité par défaut.

PRECISION

Paramètre facultatif indiquant le niveau de précision après le séparateur décimal pour la distance renvoyée. Si vous spécifiez une valeur de précision, la valeur de la distance renvoyée est tronquée en fonction du nombre de décimales indiqué. La valeur maximale est 5. Si vous omettez cette valeur, les décimales ne sont pas tronquées.

## Description

DISTANCE calcule la distance entre deux points. La latitude et la longitude doivent être exprimées sous forme d'unités décimales. Utilisez toujours une virgule et un espace pour séparer les valeurs numériques. Cela est obligatoire pour les langages utilisant une virgule comme séparateur décimal, comme le montre le deuxième exemple ci-dessous.

## Exemples

DISTANCE (18.529747, 73.839798, 18.533511, 73.8777995, MILES, 2) renvoie la valeur de 2.50 miles.

DISTANCE (18,529747, 73,839798, 18,533511, 73,8777995, KILOMETERS, 1) renvoie la valeur de 4,0 kilomètres.

---

## Macro DIV

est disponible dans IBM Campaign et IBM Interact. DIV

## Syntaxe

data DIV divisor data / divisor

## Paramètres

data

Valeurs numérique dans lesquelles la division doit être effectuée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

divisor

Valeurs par lesquelles les valeurs de la plage de données spécifiée doivent être divisées. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans data2 doit correspondre au nombre de colonnes dans data1, à moins que data2 ne soit une constante. Pour la définition du format de divisor (identique à data), voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

DIV divise la plage de données spécifiée par la valeur du diviseur. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 divisée par la colonne correspondante de data2 (à savoir, la première colonne de data1 est divisée par la première colonne de data, la deuxième colonne à la deuxième colonne et ainsi de suite).

Si data2 est une constante, chaque valeur de data1 est divisée par cette valeur. Si data2 contient une ou plusieurs colonnes, les calculs sont effectués ligne par ligne entre une colonne de data1 et une colonne de data2. La première ligne de data1 est divisée par la première ligne de data2, la seconde ligne, par la seconde ligne, etc. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** L'utilisation d'une colonne contenant le même nombre  $x$  dans chaque ligne que divisor revient à utiliser la constante  $x$  pour divisor.

**Remarque :** L'opérateur DIV peut être abrégé en barre oblique (/).

## Exemples

$TEMP = 8 \text{ DIV } 4$ ou $TEMP = 8/4$ Crée une colonne nommée TEMP contenant la valeur deux.
$TEMP = V1/8$ Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au contenu de la colonne V1 divisé par huit.
$TEMP = V1:V3/2$ Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent au contenu de la colonne V1 divisé par deux, les valeurs de la colonne VX correspondent au contenu de la colonne V2 divisé par deux et les valeurs de la colonne VY correspondent au contenu de la colonne V3 divisé par deux.
$TEMP = V1/V1$ Crée une colonne nommée TEMP ne contenant que des uns (tout nombre divisé par lui-même étant égal à un).
$TEMP = V1/V2$ Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 divisée par la valeur de ligne correspondante de la colonne V2.
$TEMP = V1:V3/V4:V6$ Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 divisées par les valeurs de ligne correspondantes de la colonne V4. La colonne VX contient la division de la colonne V2 par V5. La colonne VY contient la division de la colonne V3 par V6.
$TEMP = V1[10:20] / V2$ ou $TEMP = V1[10:20] / V2[1:11]$ Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent le résultat de la division des valeurs des lignes 10 à 20 de la colonne V1 par les valeurs des lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
MOD	Calcule le modulo du contenu de la plage de données spécifiée
MULT	Multiplie le contenu de deux plages de données
POW	Calcule une valeur de base élevée aux puissances exponentielles spécifiées

---

## Macro EQ

est disponible dans IBM Campaign et IBM Interact. EQ

### Syntaxe

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

### Paramètres

data1

Plage de cibles à comparer. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

data2

Nombres auxquels toutes les valeurs de la colonne spécifiée doivent être comparées. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans data2 doit correspondre au nombre de colonnes dans data1, à moins que data2 ne soit une constante. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

EQ compare les deux plages de données spécifiées, puis renvoie un un si les valeurs sont égales ou un zéro si elles ne le sont pas. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 comparée à la colonne correspondante de data2 (à savoir, la première colonne de data1 est comparée à la première colonne de data, la deuxième colonne à la deuxième colonne et ainsi de suite).

Si data2 est une constante, chaque valeur de data est comparée à cette valeur. Si data2 est une colonne, les calculs sont effectués ligne par ligne. Les valeurs de data1 sont comparées à la valeur de la première ligne de data2, la deuxième ligne à la deuxième ligne et ainsi de suite. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

Lors de la comparaison de chaînes, la casse n'est pas importante ("Oui", "OUI", "oui" et "oui" sont considérées comme étant égales).

**Remarque :** L'opérateur EQ peut être abrégé en double signe égal (==). A l'intérieur des parenthèses, un signe égal unique (=) peut également être utilisé pour la fonction macro EQ (en dehors des parenthèses, le signe égal est interprété comme l'opérateur d'affectation).

## Exemples

TEMP = 3 EQ 4, TEMP = 3==4 ou TEMP = (3=4)
Crée une colonne nommée TEMP contenant la valeur zéro (car trois n'est pas égal à quatre).
TEMP = "No" == "NO"
Crée une colonne nommée TEMP contenant la valeur un (la casse n'est pas importante dans les comparaisons de chaînes).
TEMP = V1 == 8
Crée une colonne nommée TEMP, dans laquelle chaque valeur est égale à un si la ligne correspondante de la colonne V1 est égale à huit ; sinon, la valeur est égale à zéro.
TEMP = V1==V1
Crée une colonne nommée TEMP ne contenant que des uns (car tout nombre est égal à lui-même).
TEMP = V1==V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 comparée à la valeur de ligne correspondante de la colonne V2.
TEMP = V1:V3 == V4:V6
Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 comparées aux valeurs de ligne correspondantes de la colonne V4. La colonne VX compare les colonnes V2 et V5. La colonne VY compare les colonnes V3 et V6.
TEMP = V1[10:20] == V2 ou TEMP = V1[10:20] == V2[1:11]
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les résultats de la comparaison des valeurs des lignes 10 à 20 de la colonne V1 aux lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
EQ	Renvoie TRUE si une plage de données est égale à une autre
GE	Renvoie TRUE si une plage de données est supérieure ou égale à une autre
GT	Renvoie TRUE si une plage de données est supérieure à une autre
LE	Renvoie TRUE si une plage de données est inférieure ou égale à une autre
LT	Renvoie TRUE si une plage de données est inférieure à une autre
NE	Renvoie TRUE si une plage de données n'est pas égale à une autre

---

## Macro EXP

est disponible uniquement dans IBM Campaign. EXP

### Syntaxe

EXP(data)

### Paramètres

data

Valeurs numériques utilisées comme exposant du nombre naturel  $e$ . Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

EXP élève le nombre naturel  $e$  de chacune des valeurs de la plage de données spécifiée (il calcule  $e^x$ ). La constante  $e$  est égale à 2,7182818. EXP renvoie une nouvelle colonne par colonne d'entrée, chacune contenant le résultat  $e^x$  pour chaque valeur  $x$  des colonnes en entrée correspondantes. EXP est l'inverse de la fonction macro LN.

**Remarque :** Si la valeur  $x$  est trop faible, une erreur de dépassement est renvoyée. Cela se produit si  $e^x$  dépasse la valeur à virgule flottante sur 32 bits maximale ou minimale.

### Exemples

TEMP = EXP(2) Crée une colonne nommée TEMP contenant la valeur 7.39.
TEMP = EXP(V1) Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond à l'élévation de $e$ au contenu de la colonne V1.
TEMP = EXP(V1:V3) Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent à l'élévation de $e$ au contenu de la colonne V1, les valeurs de la colonne VX correspondent à l'élévation de $e$ au contenu de la colonne V2 et les valeurs de la colonne VY correspondent à l'élévation de $e$ au contenu de la colonne V3.
TEMP = EXP(V1[10:20]) Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les résultats de l'élévation de $e$ aux valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.



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

Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux résultats de l'élevation de  $e$  aux valeurs de ligne correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux résultats de l'élevation de  $e$  aux valeurs de ligne correspondantes de la colonne V2.

## Fonctions associées

Fonction	Description
LN ou LOG	Calcule le logarithme naturel du contenu de la plage de données spécifiée
LOG2	Calcule le logarithme base 2 du contenu de la plage de données spécifiée
LOG10	Calcule le logarithme base 10 du contenu de la plage de données spécifiée
POW	Puissance exponentielle

---

## Macro EXTERNALCALLOUT

La macro EXTERNALCALLOUT est disponible uniquement dans IBM Interact.

### Syntaxe

```
EXTERNALCALLOUT( calloutName, arg1, ...)
```

### Paramètres

*calloutName*

Nom de l'appel que vous avez créé à l'aide de l'API ExternalCallout. Ce nom doit correspondre au nom de la catégorie d'appels externes que vous avez créée dans la plateforme marketing IBM.

*arg1*

Argument requis par votre appel, si nécessaire.

### Description

EXTERNALCALLOUT permet d'appeler une application externe pour ajouter des données à votre diagramme temps réel. EXTERNALCALLOUT peut renvoyer tout ce pourquoi l'appel a été créé. Vous devez écrire cet appel dans Java à l'aide de l'API ExternalCallout. Pour plus d'informations, reportez-vous au document *IBM Interact - Guide d'administration*.

### Exemples

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

Appelle l'appel `getStockPrice` en transmettant le nom de l'action, `UNCA`, comme argument. Cet appel défini par l'utilisateur renvoie le cours de l'action tel qu'ils est défini par l'appel.

---

## Macro FACTORIAL

est disponible uniquement dans IBM Campaign. FACTORIAL

### Syntaxe

FACTORIAL(data)

### Paramètres

data

Valeurs entières dont la factorielle doit être calculée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs, mais la valeur doit être supérieure ou égale à zéro. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

FACTORIAL calcule la factorielle des valeurs de la plage de données spécifiée. Toutes les entrées doivent être des entiers supérieurs ou égaux à zéro. La factorielle d'un nombre entier inférieur ou égal à un est un. Pour les entiers  $x \geq 2$ , la factorielle  $x! = x(x-1)(x-2)\dots(x-(x-1))$ . FACTORIAL renvoie une nouvelle colonne par colonne en entrée, chacune contenant la factorielle des nombres de la colonne d'entrée correspondante.

**Remarque :** Toute valeur supérieure à 34 génère ??? (erreur de dépassement des valeurs à virgule flottante).

### Exemples

TEMP = FACTORIAL(3) Crée une colonne nommée TEMP contenant la valeur 6.
TEMP = FACTORIAL(-2) Génère une erreur 333, indiquant que l'argument doit être supérieur ou égal à 0.
TEMP = FACTORIAL(V1) Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond à la factorielle du contenu de la colonne V1.
TEMP = FACTORIAL(V1:V3) Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux factorielles du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux factorielles du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux factorielles du contenu de la colonne V3.
TEMP = FACTORIAL(V1[10:20]) Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les factorielles des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.

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

Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 50 (les autres cibles sont vides). Les valeurs de la colonne TEMP sont les factorielles des lignes de la colonne V1 et les valeurs de la colonne VX sont les factorielles des valeurs de la colonne V2.

---

## Macro FLOOR

est disponible uniquement dans IBM Campaign. FLOOR

### Syntaxe

FLOOR(data)

### Paramètres

data

Valeurs numériques dont la partie entière doit être calculée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

FLOOR calcule la partie entière des valeurs de la plage de données spécifiée. La partie entière d'un nombre correspond au plus grand entier inférieur à ce nombre. FLOOR renvoie une nouvelle colonne par colonne d'entrée, chacune contenant la partie entière des nombres de la colonne d'entrée correspondante.

**Remarque :** Cela est équivalent à la fonction macro INT.

### Exemples

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

Crée une colonne nommée TEMP contenant la valeur 4.

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

Crée une colonne nommée TEMP contenant la valeur -3.

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

Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond à la partie entière du contenu de la colonne V1.

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

Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux parties entières du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux parties entières du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux parties entières du contenu de la colonne V3.

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

Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les parties entières des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.

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

Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 50 (les autres cibles sont vides). Les valeurs de la colonne TEMP sont les parties entières des lignes de la colonne V1 et les valeurs de la colonne VX sont les parties entières des valeurs de la colonne V2.

## Fonctions associées

Fonction	Description
CEILING	Calcule le plafond de chaque valeur de la plage de données spécifiée
FRACTION	Renvoie la partie décimale de chaque valeur de la plage de données spécifiée
TRUNCATE	Renvoie la partie non décimale de chaque valeur de la plage de données spécifiée

---

## Macro FORMAT

est disponible uniquement dans IBM Campaign. FORMAT

### Syntaxe

Format possède deux formes ; l'une pour les données de type numérique et l'autre pour celles de type texte/caractère.

Pour les données de type numérique :

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

Pour les données de type texte/caractère :

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

### Paramètres

colName

La macro examine colName, détermine son type de données, puis impose les règles appropriées pour les paramètres suivants en conséquence.

width

La largeur doit être assez importante pour contenir le résultat complet ; sinon le résultat est tronqué. Les valeurs admises sont comprises entre 1 et 29 si colName est numérique et entre 1 et 255 dans le cas contraire.

precision

La précision est le nombre de chiffres après le séparateur décimal. Les valeurs admises sont comprises entre 0 et 15. Si la valeur est égale à zéro, le résultat est un entier. La précision par défaut est de 2.

format\_type

Les mots clés valides de format\_type sont les suivants :

PERIOD	Le point (.) est utilisé comme symbole décimal. Aucun symbole de groupement des chiffres n'est utilisé. Il s'agit de la valeur par défaut.
COMMA	La virgule (,) est utilisée comme symbole décimal. Aucun symbole de groupement des chiffres n'est utilisé.
PERIOD_COMMA	Le point est utilisé comme symbole décimal et la virgule comme symbole de groupement des chiffres.
COMMA_PERIOD	Virgule comme symbole décimal et point comme symbole de groupement des chiffres.

alignment

Les mots clés valides pour l'alignement sont LEFT et RIGHT. La valeur par défaut est RIGHT pour les données de type numérique et LEFT pour les données de type texte/caractère.

padding

Les mots clés valides pour le remplissage sont SPACE et ZERO. La valeur par défaut est SPACE. ZERO est ignoré (SPACE est utilisé à la place) si l'alignement est LEFT.

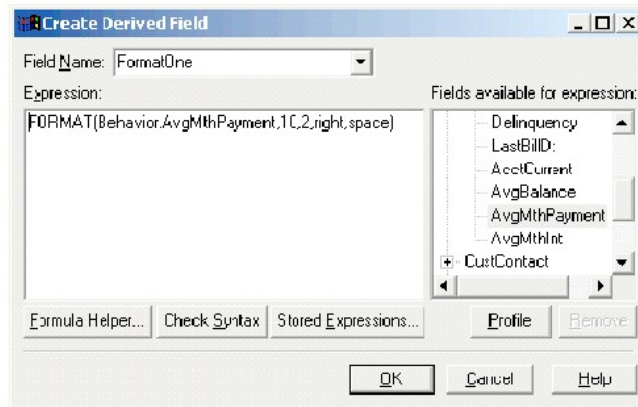
Notez que les chaînes numériques contenues dans des données de type texte/caractère sont traitées comme texte/caractère. Notez également que la forme numérique accepte plusieurs mots clés facultatifs possédant chacun une valeur par défaut. Toutefois, pour remplacer la valeur par défaut du deuxième mot clé facultatif ou des suivants, vous DEVEZ coder les valeurs par défaut des mots clés facultatifs précédents (dans les faits, ils deviennent obligatoires). Par exemple : pour remplacer l'alignement par un alignement à gauche (LEFT), vous devez coder : `FORMAT(myNumCol, 10, 2, PERIOD, LEFT)`.

## Description

FORMAT convertit des données numériques en chaîne avec diverses options de formatage pour contrôler et définir la chaîne en sortie. Cela est particulièrement utile pour créer des fichiers d'instantané avec des formats spécifiques à des fins de publipostage.

## Exemples

L'exemple ci-après définit une zone dérivée à l'aide de FORMAT.



Les exemples ci-après montrent la même zone, AvgMthPayment, dans trois formats.

Non formaté :

The screenshot shows a table with the title "UnFormatted...". The table has two columns: an ID column and a value column. The values are displayed in their raw, unformatted state.

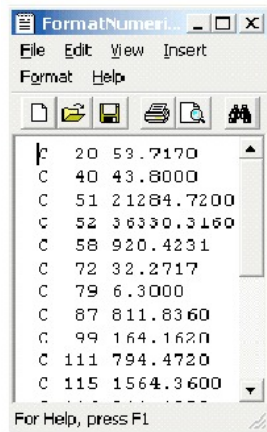
ID	AvgMthPayment
20	53.717
40	43.8
51	21284.72
52	36330.316
58	920.423123
72	32.271717
79	6.3
87	811.836
99	164.162
111	794.472
115	1564.36

Formaté à l'aide de FORMAT(Behavior.AvgMthPayment,10,2,right.space) :

The screenshot shows a table with the title "FormatNumeric...". The table has two columns: an ID column and a value column. The values are formatted according to the expression: 10 total digits, 2 decimal places, right-aligned.

ID	AvgMthPayment
20	53.72
40	43.80
51	21284.72
52	36330.32
58	920.42
72	32.27
79	6.30
87	811.84
99	164.16
111	794.47
115	1564.36

Formaté à l'aide de FORMAT(Behavior.AvgMthPayment,10,4) :



---

## Macro FRACTION

est disponible uniquement dans IBM Campaign. FRACTION

### Syntaxe

FRACTION(data)

### Paramètres

data

Valeurs numériques dont la fraction doit être calculée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

FRACTION calcule la partie décimale des valeurs de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant la partie décimale des nombres de la colonne d'entrée correspondante.

**Remarque :** Les fonctions macro FRACTION et TRUNCATE sont complémentaires car leur somme correspond aux valeurs d'origine.

### Exemples

TEMP = FRACTION(4.3)
Crée une colonne nommée TEMP contenant la valeur 0.3.
TEMP = FRACTION(2.9)
Crée une colonne nommée TEMP contenant la valeur -0.9.
TEMP = FRACTION(V1)
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond à la partie décimale du contenu de la colonne V1.

<p>TEMP = FRACTION(V1:V3)</p> <p>Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux parties décimales du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux parties décimales du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux parties décimales du contenu de la colonne V3.</p>
<p>TEMP = FRACTION(V1[10:20])</p> <p>Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les parties décimales des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.</p>
<p>TEMP = FRACTION(V1[50:99]:V2)</p> <p>Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 50 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux parties décimales des lignes de la colonne V1 et les valeurs de la colonne VX correspondent aux parties décimales des valeurs de la colonne V2.</p>

## Fonctions associées

Fonction	Description
CEILING	Calcule le plafond de chaque valeur de la plage de données spécifiée
FLOOR	Calcule la partie entière de chaque valeur de la plage de données spécifiée
TRUNCATE	Renvoie la partie non décimale de chaque valeur de la plage de données spécifiée

---

## Macro GE

La macro GE est disponible dans IBM Campaign and IBMInteract.

### Syntaxe

data1 GE data2 data1 >= data2

### Paramètres

data1

Plage de cibles numériques à comparer. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

data2

Nombres auxquels toutes les valeurs de la colonne spécifiée doivent être comparées. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans data2 doit correspondre au nombre de colonnes dans data1, à moins que data2 ne soit une constante. Pour la définition du format de data, voir la



section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

GE compare les deux plages de données spécifiées, puis renvoie un un si les valeurs du premier jeu de données sont supérieures ou égales à celles du deuxième jeu de données ou un zéro dans le cas contraire. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 comparée à la colonne correspondante de data2 (à savoir, la première colonne de data1 est comparée à la première colonne de data, la deuxième colonne à la deuxième colonne et ainsi de suite).

Si data2 est une constante, chaque valeur de data est comparée à cette valeur. Si data2 est une colonne, les calculs sont effectués ligne par ligne. Les valeurs de data1 sont comparées à la valeur de la première ligne de data2, la deuxième ligne à la deuxième ligne et ainsi de suite. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** L'opérateur GE peut être abrégé en signe supérieur suivi d'un signe égal (>=).

## Exemples

<p>TEMP = 9 GE 4 ou TEMP = 9 &gt;= 4</p> <p>Crée une colonne nommée TEMP contenant la valeur un (car neuf est supérieur à quatre).</p>
<p>TEMP = V1 &gt;= 8</p> <p>Crée une colonne nommée TEMP, dans laquelle chaque valeur est égale à un si la ligne correspondante de la colonne V1 est supérieure ou égale à huit ; sinon, la valeur est égale à zéro.</p>
<p>TEMP = V1:V3 &gt;= 2</p> <p>Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent au contenu de la colonne V1 comparé à la valeur deux, les valeurs de la colonne VX correspondent au contenu de la colonne V2 comparé à la valeur deux, et les valeurs de la colonne VY correspondent au contenu de la colonne V3 comparé à la valeur deux.</p>
<p>TEMP = V1 &gt;= V1</p> <p>Crée une colonne nommée TEMP ne contenant que des uns (car tout nombre est égal à lui-même).</p>
<p>TEMP = V1 &gt;= V2</p> <p>Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 comparée à la valeur de ligne correspondante de la colonne V2.</p>
<p>TEMP = V1:V3 &gt;= V4:V6</p> <p>Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 comparées aux valeurs de ligne correspondantes de la colonne V4. La colonne VX compare les colonnes V2 et V5. La colonne VY compare les colonnes V3 et V6.</p>
<p>TEMP = V1[10:20] &gt;= V2 ou TEMP = V1[10:20] &gt;= V2[1:11]</p> <p>Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les résultats de la comparaison des valeurs des lignes 10 à 20 de la colonne V1 aux valeurs des lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.</p>

## Fonctions associées

NE	Renvoie TRUE si une plage de données n'est pas égale à une autre
----	--

---

## Macro GROUPBY

est disponible uniquement dans IBM Campaign. GROUPBY

### Syntaxe

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

### Paramètres

- `group_field`  
Indique la variable sur laquelle les enregistrements sont regroupés (toutes les valeurs identiques de la variable spécifiée sont regroupées ensemble).
- `keyword`  
Indique la fonction de cumul récapitulatif à effectuer sur `rolled-field`.
- `rolled_field`  
Indique la variable à récapituler ou cumuler.
- `output_field`  
Identifie une variable de remplacement à renvoyer pour une ligne d'un groupe et ne pouvant être utilisée qu'avec les mots clés `MinOf`, `MaxOf` et `MedianOf`.

### Description

GROUPBY récapitule plusieurs lignes de données dans un groupe. La sortie de cette fonction est une colonne unique. La sortie est le résultat de l'opération spécifiée par `keyword` sur la zone `rolled_field` du groupe homogène spécifié par `group_field`. S'il existe plusieurs réponses correspondant à une condition spécifiée, la première détectée est renvoyée.

Si le paramètre facultatif `output_field` n'est pas fourni, la sortie est le résultat de l'opération sur `rolled_field`. Si `output_field` est fourni, le résultat est la zone `output_field` de la ligne dans le groupe.

Si plusieurs lignes d'un groupe répondent à la condition spécifiée (par exemple, il existe des liens pour la valeur max), la zone `output-field` associé à la première ligne remplissant la condition est renvoyé.

**Remarque :** Pour effectuer un groupement sur plusieurs colonnes, vous pouvez placer entre accolades "{ }" une liste de noms de zone, séparés par des virgules, et utiliser cette liste comme premier paramètre de l'appel de macro GROUPBY.

Les mots clés pris en charge sont les suivants (casse indifférente) :

Mot clé	Chaîne ?	Description
	Oui/Non	
CountOf	Oui	Renvoie le nombre d'enregistrements de chaque groupe (rolled_field peut être de type numérique ou chaîne ; la valeur renvoyée est la même quelle que soit la valeur de rolled_field).
MinOf	Oui	Renvoie la valeur minimale de rolled_field dans chaque groupe (rolled_field peut être de type numérique ou chaîne ; si rolled_field est une chaîne, la valeur la plus proche du début de l'alphabet (pour un tri alphabétique) est renvoyée).
MaxOf	Oui	Renvoie la valeur maximale de rolled_field dans chaque groupe (rolled_field peut être de type numérique ou chaîne ; si rolled_field est une chaîne, la valeur la plus proche de la fin de l'alphabet (pour un tri alphabétique) est renvoyée).
DiffOf	Oui	Renvoie le nombre de valeurs distinctes de rolled_field dans chaque groupe (rolled_field peut être de type numérique ou chaîne).
AvgOf	Non	Renvoie la valeur moyenne de rolled_field dans chaque groupe (rolled_field doit être numérique).
ModeOf	Oui	Renvoie la valeur modale (à savoir, la valeur la plus fréquente) de rolled_field dans chaque groupe (rolled_field peut être de type numérique ou chaîne).
MedianOf	Oui	Renvoie la valeur médiane (à savoir, la valeur du milieu lors d'un tri en fonction de rolled_field) de rolled_field dans chaque groupe (rolled_field peut être de type numérique ou chaîne ; si rolled_field est une chaîne, les valeurs sont triées par ordre alphabétique).

<b>Mot clé</b>	<b>Chaîne ? Oui/Non</b>	<b>Description</b>
OrderOf	Oui	Renvoie l'ordre de rolled_field dans chaque groupe (rolled_field doit être numérique). Si plusieurs enregistrements ont la même valeur, ils reçoivent tous la même valeur.
SumOf	Non	Renvoie la somme de rolled_field dans chaque groupe (rolled_field doit être numérique).
StdevOf	Non	Renvoie l'écart type de rolled_field dans chaque groupe (rolled_field doit être numérique).
IndexOf	Oui	Renvoie l'index de base 1 (trié par rolled_field) de chaque enregistrement (rolled_field peut être de type numérique ou chaîne). L'ordre de tri est croissant.  Remarque : pour les zones numériques, l'ordre de tri de RankOf et IndexOf peut être décroissant, si l'on place un signe moins (-) devant la zone de tri.
RankOf	Oui	Renvoie la catégorie de base 1 (triée par rolled_field) dans laquelle se trouve chaque enregistrement (rolled_field peut être de type numérique ou chaîne). L'ordre de tri est croissant.  Remarque : pour les zones numériques, l'ordre de tri de RankOf et IndexOf peut être décroissant, si l'on place un signe moins (-) devant la zone de tri.

## Exemples

GROUPBY (Household\_ID, SumOf, Account\_Balance)

Calcule la somme de tous les soldes de compte par ménage.

GROUPBY (Cust\_ID, MinOf, Date(Account\_Open\_Date), Acc\_Num)

Renvoie le numéro de compte du premier compte ouvert par le client.

---

## Macro GROUPBY\_WHERE

est disponible uniquement dans IBM Campaign. GROUPBY\_WHERE

### Syntaxe

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

### Paramètres

- `group_field`  
Indique la variable sur laquelle les enregistrements sont regroupés (toutes les valeurs identiques de la variable spécifiée sont regroupées ensemble).
- `keyword`  
Indique la fonction de cumul récapitulatif à effectuer.
- `rolled_field`  
Indique la variable à récapituler ou cumuler.
- `where_value`  
Expression convertie en un ou zéro qui indique les lignes à inclure dans l'opération de cumul.
- `output_field`  
Identifie une variable de remplacement à renvoyer pour une ligne d'un groupe et ne pouvant être utilisée qu'avec les mots clés `MinOf`, `MaxOf` et `MedianOf`

### Description

GROUPBY\_WHERE récapitule des lignes de données spécifiques dans un groupe. La sortie de cette fonction est une colonne unique. La sortie est le résultat de l'opération spécifiée par `keyword` sur la zone `rolled_field` du groupe homogène spécifié par `group_field` et filtré par `where_value`. Seules les lignes pour lesquelles `where_value` contient la valeur un sont incluses dans le calcul.

Si le paramètre facultatif `output_field` n'est pas fourni, la sortie est le résultat de l'opération sur `rolled_field`. Si `output_field` est fourni, le résultat est la zone `output_field` de la ligne dans le groupe.

**Remarque :** Pour plus d'informations sur les valeurs valides de `keyword`, voir «Macro GROUPBY», à la page 76.

### Exemples

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

Calcule la somme de tous les comptes dont les soldes sont positifs pour chaque ménage.

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

Calcule le montant des achats moyen de chaque client pour les achats au cours des 90 derniers jours.

---

## Macro GT

est disponible dans IBM Campaign et IBM Interact. GT

## Syntaxe

data1 GT data2 data1 > data2

## Paramètres

data1

Plage de cibles numériques à comparer. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

data2

Nombre auxquels toutes les valeurs de la colonne spécifiée doivent être comparées. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans data2 doit correspondre au nombre de colonnes dans data1, à moins que data2 ne soit une constante. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

GT compare les deux plages de données spécifiées, puis renvoie un un si les valeurs du premier jeu de données sont supérieures à celles du deuxième jeu de données ou un zéro dans le cas contraire. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 comparée à la colonne correspondante de data2 (à savoir, la première colonne de data1 est comparée à la première colonne de data, la deuxième colonne à la deuxième colonne et ainsi de suite).

Si data2 est une constante, chaque valeur de data est comparée à cette valeur. Si data2 est une colonne, les calculs sont effectués ligne par ligne. Les valeurs de data1 sont comparées à la valeur de la première ligne de data2, la deuxième ligne à la deuxième ligne et ainsi de suite. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** L'opérateur GT peut être abrégé en signe supérieur (>).

## Exemples

```
TEMP = 3 GT 4 ou TEMP = 3 > 4
```

Crée une colonne nommée TEMP contenant la valeur zéro (car trois n'est pas supérieur à quatre).

```
TEMP = V1 > 8
```

Crée une colonne nommée TEMP, dans laquelle chaque valeur est égale à un si la ligne correspondante de la colonne V1 est supérieure à huit ; sinon, la valeur est égale à zéro.

TEMP = V1:V3 > 2
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent au contenu de la colonne V1 comparé à la valeur deux, les valeurs de la colonne VX correspondent au contenu de la colonne V2 comparé à la valeur deux, et les valeurs de la colonne VY correspondent au contenu de la colonne V3 comparé à la valeur deux.
TEMP = V1 > V1
Crée une colonne nommée TEMP ne contenant que des zéros (car aucun nombre n'est supérieur à lui-même).
TEMP = V1 > V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 comparée à la valeur de ligne correspondante de la colonne V2.
TEMP = V1:V3 > V4:V6
Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 comparées aux valeurs de ligne correspondantes de la colonne V4. La colonne VX compare les colonnes V2 et V5. La colonne VY compare les colonnes V3 et V6.
TEMP = V1[10:20] > V2 ou TEMP = V1[10:20] > V2[1:11]
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les résultats de la comparaison des valeurs des lignes 10 à 20 de la colonne V1 aux valeurs des lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
EQ	Renvoie TRUE si une plage de données est égale à une autre
GE	Renvoie TRUE si une plage de données est supérieure ou égale à une autre
LE	Renvoie TRUE si une plage de données est inférieure ou égale à une autre
LT	Renvoie TRUE si une plage de données est inférieure à une autre
NE	Renvoie TRUE si une plage de données n'est pas égale à une autre

---

## Macro IF

est disponible dans IBM Campaign et IBM Interact. IF

### Syntaxe

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

### Paramètres

predicate\_col

Colonne de valeurs booléennes ou expression convertie en une colonne de valeurs booléennes. Les valeurs booléennes sont interprétées comme des nulles ou non

nulles. Cette colonne doit contenir au moins autant de lignes que la plage de données de laquelle les données sont extraites.

then\_value

Valeurs à renvoyer si la ligne de predicate\_col correspondante contient une valeur non nulle. Il peut s'agir d'une valeur constante, d'une colonne ou d'une expression convertie comme l'une de ces valeurs. Pour la définition de format de then\_value (identique à data), voir «Paramètres des fonctions macro pour IBM Campaign», à la page 8.

else\_value

Si ce paramètre facultatif est fourni, il est renvoyé si la ligne correspondante de predicate\_col contient un zéro. Il peut s'agir d'une valeur constante, d'une colonne ou d'une expression convertie comme l'une de ces valeurs. Si else\_value n'est pas fourni, un zéro est renvoyé chaque fois que predicate\_col a la valeur false. Pour la définition de format de else\_value (identique à data), voir «Paramètres des fonctions macro pour IBM Campaign», à la page 8.

## Description

IF convertit l'expression de predicate\_col et renvoie then\_value si l'expression est true ou else\_value si l'expression est false. Il renvoie le même nombre de colonnes dans then\_value et else\_value. Les nouvelles colonnes contiendront les valeurs de ligne correspondantes de then\_value si la valeur de predicate\_col est non nulle. Si else\_value est fourni, il est renvoyé lorsque la valeur de predicate\_col est zéro. Si else\_value n'est pas fourni, zéro est renvoyé.

IF fonctionnant ligne par ligne, il génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte (c'est-à-dire la colonne la plus courte entre predicate\_col, then\_value et else\_value).

**Remarque :** En général, une colonne de prédicat est créée à l'aide de l'une des fonctions macro de comparaison (par exemple, ==, >, <, ISEVEN, ISODD, etc.).

## Exemples

TEMP = IF(1, V1)Crée une colonne nommée TEMP contenant une copie de la colonne V1.
TEMP = IF(V1, 1, 0) Crée une colonne nommée TEMP, dans laquelle chaque valeur est égale à un si la valeur correspondante de la colonne V1 est non nulle ; sinon, la valeur est égale à zéro.
TEMP = IF(V3, V1, V2) Crée une colonne nommée TEMP, dans laquelle chaque valeur est copiée à partir de la colonne V1 si la valeur correspondante de la colonne V3 est non nulle ; sinon; la valeur est copiée à partir de la colonne V2.
TEMP = IF(ABS(V1-AVG(V1)) < STDV(V1), V1) Crée une colonne nommée TEMP contenant chaque valeur de la colonne V1 qui est inférieure d'un écart type de la moyenne.



```
TEMP = IF(V3[20:30], V1[30:40], V2)
```

Crée une colonne nommée TEMP contenant les valeurs des lignes 10 à 20. Chaque valeur est copiée à partir de la colonne V1 (cibles 10 à 20) si la valeur correspondante de la colonne V3 (cibles 30 à 40) est non nulle ; sinon, la valeur est copiée à partir de la colonne V2 (cibles 1 à 11).

---

## Macro IN

est disponible dans IBM Campaign et IBM Interact. IN

### Syntaxe

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

### Paramètres

La première forme autorise l'utilisation d'une liste de valeurs à la place d'une sous-requête.

La seconde forme utilise une sous-requête qui est convertie pour générer un résultat intermédiaire, sur lequel un traitement supplémentaire peut être effectué.

### Description

Le prédicat IN permet d'utiliser une liste de valeurs au lieu d'une sous-requête ou lance une sous-requête.

**Remarque :** Le prédicat IN possède une version négative : NOT IN. Son format est identique à celui de IN. NOT IN est vrai uniquement si la valeur fournie est introuvable dans les valeurs renvoyées par la sous-requête.

**Important :** Si vous utilisez IN dans IBM Interact, vous ne pouvez utiliser que la syntaxe value IN (value1 AND value2 . . . .).

### Exemples

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

Renvoie les colonnes spécifiées d'une plage de données

```
TEMP = IN("cat", COLUMN("cat", "dog", "bird"))
```

Crée une colonne nommée TEMP contenant la valeur un.

```
TEMP = IN(V1, V1)
```

Crée une colonne nommée TEMP ne contenant que des uns.

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

Crée une colonne nommée TEMP, dans laquelle chaque valeur est égale à un si la ligne correspondante de la colonne V1 contient une valeur dans la colonne V2 et à zéro dans le cas contraire.

---

## Macro INT

est disponible uniquement dans IBM Campaign. INT

## Syntaxe

INT(data)

## Paramètres

data

Valeurs numériques à arrondir en valeurs entières. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

INT calcule l'entier le plus grand inférieur aux valeurs (également appelé partie entière) de la plage de données spécifiée. INT renvoie une nouvelle colonne par colonne d'entrée, chacune contenant la partie entière des nombres de la colonne d'entrée correspondante.

**Remarque :** Cela est équivalent à la fonction macro FLOOR.

## Exemples

TEMP = INT(4.7)
Crée une colonne nommée TEMP contenant la valeur 4.
TEMP = INT(-1.5)
Crée une colonne nommée TEMP contenant la valeur -2.
TEMP = INT(V1)
Crée une colonne nommée TEMP, où chaque valeur correspond à l'entier le plus grand inférieur ou égal au contenu de la colonne V1.
TEMP = V1 - INT(V1)
Crée une colonne nommée TEMP contenant la partie décimale de chaque valeur de la colonne V1.
TEMP = INT(V1:V3)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux entiers les plus grands inférieurs ou égaux au contenu de la colonne V1, les valeurs de la colonne VX correspondent aux entiers les plus grands inférieurs ou égaux au contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux entiers les plus grands inférieurs ou égaux au contenu de la colonne V3.
TEMP = INT(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les entiers les plus grands inférieurs ou égaux aux valeurs correspondantes des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.

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

Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux entiers les plus grands inférieurs ou égaux aux valeurs de ligne correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux entiers les plus grands inférieurs ou égaux aux valeurs de ligne correspondantes de la colonne V2.

## Fonctions associées

Fonction	Description
ROUND	Calcule la valeur arrondie du contenu de la plage de données spécifiée
TRUNCATE	Renvoie la partie non décimale de chaque valeur de la plage de données spécifiée

---

## Macro INVERSE

est disponible uniquement dans IBM Campaign. INVERSE

### Syntaxe

INVERSE(data)

### Paramètres

data

Valeurs numériques dont l'inverse doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

INVERSE calcule la valeur négative des valeurs de la plage de données spécifiée. Il renvoie  $-x$  (les valeurs négatives sont renvoyées comme valeurs positives et les valeurs positives sont renvoyées comme valeurs négatives). INVERSE renvoie une nouvelle colonne par colonne d'entrée, chacune contenant l'inverse des valeurs de la colonne d'entrée correspondante.

**Remarque :** Pour inverser une valeur ou une colonne, précédez-la d'un signe moins (-). Par exemple,  $V2 = -V1$  est équivalent à  $V2 = \text{INVERSE}(V1)$ .

### Exemples

TEMP = INVERSE(3.2)

Crée une colonne nommée TEMP contenant la valeur -3.2.

TEMP = INVERSE(V1)

Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond à la valeur négative des valeurs de la colonne V1.

<p>TEMP = INVERSE(V1:V3)</p> <p>Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux valeurs négatives des valeurs de la colonne V1, les valeurs de la colonne VX correspondent aux valeurs négatives des valeurs de la colonne V2 et les valeurs de la colonne VY correspondent aux valeurs négatives des valeurs de la colonne V3.</p>
<p>TEMP = INVERSE(V1[10:20])</p> <p>Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les valeurs négatives des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.</p>
<p>TEMP = INVERSE(V1[1:5]:V2)</p> <p>Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux valeurs négatives des valeurs des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux valeurs négatives des valeurs des lignes correspondantes de la colonne V2.</p>

## Fonctions associées

Fonction	Description
ABS	Calcule la valeur absolue du contenu de la plage de données spécifiée
NOT	Calcule le non logique (NON) du contenu de la plage de données spécifiée
SIGN	Détermine le signe (positif ou négatif) des valeurs dans la plage de données spécifiée

---

## Macro IS

est disponible dans IBM Campaign et IBM Interact. IS

### Syntaxe

IS <keyword>

### Paramètres

keyword

Condition de recherche, généralement "NULL," "TRUE," "UNKNOWN" et "FALSE."

### Description

IS est utilisé dans les conditions de recherche complexes. Plus la recherche est complexe, plus la condition IS peut s'avérer utile. Les conditions de recherche booléenne de ce type permettent d'améliorer de simples conditions de recherche.

IS renvoie des résultats différents dans IBM Interact et IBM Campaign. NULL renvoie 1 s'il existe au moins une valeur NULL pour un ID audience. UNKNOWN renvoie 1 pour un ID audience s'il ne possède pas de valeur.

---

## Macro ISERROR

est disponible uniquement dans IBM Campaign. ISERROR

### Syntaxe

ISERROR(data)

### Paramètres

data

Valeurs à tester pour vérifier si l'une des lignes contient une erreur (à savoir, une cible ???). Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

ISERROR vérifie si l'une des cibles de chaque ligne de la plage de données spécifiée contient une erreur (à savoir, une cible ???). Il renvoie une nouvelle colonne, chaque ligne contenant un un si la ligne correspondante de data contient une erreur. Sinon, la ligne contient un zéro. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus longue.

### Exemples

TEMP = ISERROR(-3)
Crée une colonne nommée TEMP contenant la valeur zéro.
TEMP = ISERROR(V1)
Crée une colonne nommée TEMP, dans laquelle chaque valeur est égale à un si la ligne correspondante de la colonne V1 contient ??? et à zéro dans le cas contraire.
TEMP = ISERROR(V1:V3)
Crée une colonne nommée TEMP, dans laquelle chaque valeur est égale à un si l'une des cibles des lignes correspondantes des colonnes V1 à V3 contient ??? et à zéro dans le cas contraire.
TEMP = ISERROR(V1[50:100]:V10)
Crée une colonne nommée TEMP, avec les valeurs des lignes 1 à 50. Chaque valeur est égale à un si l'une des cibles des lignes 50 à 100 des colonnes V1 à V10 contient ??? et à zéro dans le cas contraire.

---

## Macro ISODD

est disponible uniquement dans IBM Campaign. ISODD

### Syntaxe

ISODD(data)

## Paramètres

data

Valeurs numériques à tester pour vérifier si elles sont impaires. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

ISODD teste chaque valeur du jeu de données spécifié pour vérifier si elle est impaire. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant un un pour toutes les valeurs impaires (à savoir, le modulo deux de la valeur est un) ou un zéro pour toutes les valeurs non impaires (paires).

**Remarque :** Pour les valeurs autres que des entiers, la fonction macro INT est appliquée au préalable. Par exemple, ISODD(2.5) = 0 car 2 n'est pas impaire.

## Exemples

TEMP = ISODD(-3)
Crée une colonne nommée TEMP contenant la valeur un.
TEMP = ISODD(V1)
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au résultat du test de valeur impaire du contenu de la colonne V1.
TEMP = ISODD(V1:V3)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux résultats du test de valeur impaire du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux résultats du test de valeur impaire du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux résultats du test de valeur impaire du contenu de la colonne V3.
TEMP = ISODD(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les résultats du test de valeur impaire des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.
TEMP = ISODD(V1[1:5]:V2)
Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux résultats du test de valeur impaire des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux résultats du test de valeur impaire des lignes correspondantes de la colonne V2.

## Fonctions associées

Fonction	Description
ISEVEN	Teste si les valeurs en entrée sont paires (à savoir, divisible par deux)

---

## Macro ISEVEN

est disponible uniquement dans IBM Campaign. ISEVEN

### Syntaxe

ISEVEN(data)

### Paramètres

data

Valeurs numériques à tester pour vérifier si elles sont paires. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

ISEVEN teste chaque valeur du jeu de données spécifié pour vérifier si elle est paire. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant un un pour toutes les valeurs paires (à savoir, le modulo deux de la valeur est zéro) ou un zéro pour toutes les valeurs non paires (impaires).

**Remarque :** Pour les valeurs autres que des entiers, la fonction macro INT est appliquée au préalable. Par exemple, ISEVEN(2.5) = 1 car 2 est paire.

### Exemples

TEMP = ISEVEN(-3)
Crée une colonne nommée TEMP contenant la valeur zéro.
TEMP = ISEVEN(V1)
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au résultat du test de valeur paire du contenu de la colonne V1.
TEMP = ISEVEN(V1:V3)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux résultats du test de valeur paire du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux résultats du test de valeur paire du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux résultats du test de valeur paire du contenu de la colonne V3.
TEMP = ISEVEN(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les résultats du test de valeur paire des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.
TEMP = ISEVEN(V1[1:5]:V2)
Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux résultats du test de valeur paire des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux résultats du test de valeur paire des lignes correspondantes de la colonne V2.

## Fonctions associées

Fonction	Description
ISODD	Teste si les valeurs en entrée sont impaires (à savoir, non divisible par deux)

---

## Macro ISODD

est disponible uniquement dans IBM Campaign. ISODD

### Syntaxe

ISODD(data)

### Paramètres

data

Valeurs numériques à tester pour vérifier si elles sont impaires. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

ISODD teste chaque valeur du jeu de données spécifié pour vérifier si elle est impaire. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant un un pour toutes les valeurs impaires (à savoir, le modulo deux de la valeur est un) ou un zéro pour toutes les valeurs non impaires (paires).

**Remarque :** Pour les valeurs autres que des entiers, la fonction macro INT est appliquée au préalable. Par exemple, ISODD(2.5) = 0 car 2 n'est pas impaire.

### Exemples

TEMP = ISODD(-3)
Crée une colonne nommée TEMP contenant la valeur un.
TEMP = ISODD(V1)
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au résultat du test de valeur impaire du contenu de la colonne V1.
TEMP = ISODD(V1:V3)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux résultats du test de valeur impaire du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux résultats du test de valeur impaire du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux résultats du test de valeur impaire du contenu de la colonne V3.
TEMP = ISODD(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les résultats du test de valeur impaire des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.



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

Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux résultats du test de valeur impaire des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux résultats du test de valeur impaire des lignes correspondantes de la colonne V2.

## Fonctions associées

Fonction	Description
ISEVEN	Teste si les valeurs en entrée sont paires (à savoir, divisible par deux)

---

## Macro LE

est disponible dans IBM Campaign et IBM Interact. LE

### Syntaxe

data1 LE data2 data1 <= data2

### Paramètres

data1

Plage de cibles numériques à comparer. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

data2

Nombres auxquels toutes les valeurs de la colonne spécifiée doivent être comparées. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

LE compare les deux plages de données spécifiées, puis renvoie un un si les valeurs du premier jeu de données sont inférieures ou égales à celles du deuxième jeu de données ou un zéro dans le cas contraire. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 comparée à la colonne correspondante de data2 (à savoir, la première colonne de data1 est comparée à la première colonne de data, la deuxième colonne à la deuxième colonne et ainsi de suite).

Si data2 est une constante, chaque valeur de data est comparée à cette valeur. Si data2 est une colonne, les calculs sont effectués ligne par ligne. Les valeurs de data1 sont comparées à la valeur de la première ligne de data2, la deuxième ligne

à la deuxième ligne et ainsi de suite. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** L'opérateur LE peut être abrégé en signe inférieur suivi d'un signe égal (<=).

## Exemples

TEMP = 4 LE 4 ou TEMP = 4 <= 4
Crée une colonne nommée TEMP contenant la valeur un (car quatre est égal à lui-même).
TEMP = V1 <= 8
Crée une colonne nommée TEMP, dans laquelle chaque valeur est égale à un si la ligne correspondante de la colonne V1 est inférieure ou égale à huit ; sinon, la valeur est égale à zéro.
TEMP = V1:V3 <= 2
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent au contenu de la colonne V1 comparé à la valeur deux, les valeurs de la colonne VX correspondent au contenu de la colonne V2 comparé à la valeur deux, et les valeurs de la colonne VY correspondent au contenu de la colonne V3 comparé à la valeur deux.
TEMP = V1 <= V1
Crée une colonne nommée TEMP ne contenant que des uns (car tout nombre est égal à lui-même).
TEMP = V1 <= V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 comparée à la valeur de ligne correspondante de la colonne V2.
TEMP = V1[10:20] <= V2 ou TEMP = V1[10:20] <= V2[1:11]
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les résultats de la comparaison des valeurs des lignes 10 à 20 de la colonne V1 aux valeurs des lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
EQ	Renvoie TRUE si une plage de données est égale à une autre
GE	Renvoie TRUE si une plage de données est supérieure ou égale à une autre
GT	Renvoie TRUE si une plage de données est supérieure à une autre
LT	Renvoie TRUE si une plage de données est inférieure à une autre
NE	Renvoie TRUE si une plage de données n'est pas égale à une autre

---

## Macro LIKE

est disponible dans IBM Campaign et IBM Interact. LIKE

## Syntaxe

data1 [NOT] LIKE data2

## Paramètres

data1

Plage de cibles à comparer. Il peut s'agir d'une chaîne de texte ou d'une expression convertie en chaîne de texte. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

data2

Masque textuel auxquels toutes les valeurs de la colonne spécifiée doivent être comparées. Il peut s'agir d'une chaîne de texte ou d'une expression convertie en chaîne de texte. Le nombre de colonnes dans data2 doit correspondre au nombre de colonnes dans data1, à moins que data2 ne soit une constante. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

Un trait de soulignement ( \_ ) dans data2 représente un caractère générique correspondant à tout caractère de data1. Un signe pourcentage (%) correspond à zéro, un ou plusieurs caractères de data1.

## Description

LIKE compare les deux plages de données spécifiées, puis renvoie un un si les chaînes correspondent ou un zéro dans le cas contraire. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 comparée à la colonne correspondante de data2 (à savoir, la première colonne de data1 est comparée à la première colonne de data2, la deuxième colonne à la deuxième colonne et ainsi de suite).

Si data2 est une constante de chaîne, chaque chaîne de data1 est comparée à cette chaîne. Si data2 est une colonne, les calculs sont effectués ligne par ligne. La chaîne de la première ligne de data1 est comparée à la chaîne de la première ligne de data2, la deuxième ligne à la deuxième ligne et ainsi de suite. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière chaîne de la colonne la plus courte.

Lors de la comparaison de chaînes, la casse n'est pas importante ("Oui", "OUI", "oui" et "ouI" sont considérées comme étant égales).

**Remarque :** La macro LIKE possède une version négative, NOT LIKE. Son format est identique à celui de LIKE. NOT LIKE renvoie un un si la chaîne de data1 ne correspond pas au modèle défini par data2.

## Exemples

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

Crée une colonne nommée TEMP contenant la valeur un (car les deux chaînes correspondent).

TEMP = "No" LIKE "NO"
Crée une colonne nommée TEMP contenant la valeur un (la casse n'est pas importante dans les comparaisons de chaînes).
TEMP = V1 LIKE "gold%"
Crée une colonne nommée TEMP, dans laquelle chaque valeur est égale à un si la ligne correspondante de la colonne V1 est égale à la chaîne "gold" suivie d'un ou de plusieurs caractères. Sinon, chaque valeur est égale à zéro.
TEMP = V1 LIKE "g_ld"
Crée une colonne nommée TEMP, dans laquelle chaque valeur est égale à un si la ligne correspondante de la colonne V1 est égale à la chaîne "g" suivie de tout caractère, lui-même suivi de "ld". Sinon, chaque valeur est égale à zéro.
TEMP = V1 LIKE V1
Crée une colonne nommée TEMP ne contenant que des uns (car tout nombre est égal à lui-même).
TEMP = V1 LIKE V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 comparée à la valeur de ligne correspondante de la colonne V2.
TEMP = V1:V3 LIKE V4:V6
Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les chaînes de V1 comparées aux chaînes de ligne correspondantes de la colonne V4. La colonne VX compare les colonnes V2 et V5. La colonne VY compare les colonnes V3 et V6.
TEMP = V1[10:20] LIKE V2 ou TEMP = V1[10:20] LIKE V2[1:11]
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les résultats de la comparaison des chaînes des lignes 10 à 20 de la colonne V1 aux lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
EQ	Renvoie TRUE si une plage de données est égale à une autre
GE	Renvoie TRUE si une plage de données est supérieure ou égale à une autre
GT	Renvoie TRUE si une plage de données est supérieure à une autre
LE	Renvoie TRUE si une plage de données est inférieure ou égale à une autre
LT	Renvoie TRUE si une plage de données est inférieure à une autre
NE	Renvoie TRUE si une plage de données n'est pas égale à une autre

---

## Macro LN ou LOG

est disponible uniquement dans IBM Campaign. LN ou LOG

## Syntaxe

LN(data) ou LOG(data)

## Paramètres

data

Valeurs numériques dont le logarithme naturel doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

LN ou LOG calcule le logarithme naturel de chaque valeur de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant le logarithme naturel des nombres de la colonne d'entrée correspondante. Les logarithmes naturels sont basés sur la constante  $e = 2.7182818$ . LN est l'inverse de la fonction macro EXP.

**Remarque :** Toutes les valeurs de la plage de données spécifiée doivent être supérieures à zéro. Dans le cas contraire, une cible vide est renvoyée pour chaque entrée non valide.

## Exemples

TEMP = LN(3) ou TEMP = LOG(3) Crée une colonne nommée TEMP contenant la valeur 1.099.
TEMP = LN(V1) Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au logarithme naturel du contenu de la colonne V1.
TEMP = LN(V1:V3) Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux logarithmes naturels du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux naturels du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux logarithmes naturels du contenu de la colonne V3.
TEMP = LN(V1[10:20]) Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les logarithmes naturels des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.
TEMP = LN(V1[1:5]:V2) Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux logarithmes naturels des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux logarithmes naturels des lignes correspondantes de la colonne V2.

## Fonctions associées

Fonction	Description
EXP	Calcule le nombre naturel (e) élevé à la puissance du contenu de chaque cible de la plage de données spécifiée
LOG2	Calcule le logarithme base 2 du contenu de la plage de données spécifiée
LOG10	Calcule le logarithme base 10 du contenu de la plage de données spécifiée
POW	Calcule une valeur de base élevée aux puissances exponentielles spécifiées

---

## Macro LOG2

est disponible uniquement dans IBM Campaign. LOG2

### Syntaxe

LOG2(data)

### Paramètres

data

Valeurs numériques dont le logarithme base 2 doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

LOG2 calcule le logarithme base 2 des valeurs de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant le logarithme base 2 des nombres de la colonne d'entrée correspondante.

**Remarque :** Toutes les valeurs de la plage de données spécifiée doivent être supérieures à zéro. Dans le cas contraire, une cible vide est renvoyée pour chaque entrée non valide.

### Exemples

TEMP = LOG2(8)
Crée une colonne nommée TEMP contenant la valeur trois.
TEMP = LOG2(V1)
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au logarithme base 2 du contenu de la colonne V1.

TEMP = LOG2(V1:V3)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux logarithmes base 2 du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux logarithmes base 2 du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux logarithmes base 2 du contenu de la colonne V3.
TEMP = LOG2(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les logarithmes base 2 des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.
TEMP = LOG2(V1[1:5]:V2)
Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux logarithmes base 2 des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux logarithmes base 2 des lignes correspondantes de la colonne V2.

## Fonctions associées

Fonction	Description
LN ou LOG	Calcule le logarithme naturel du contenu de la plage de données spécifiée
LOG10	Calcule le logarithme base 10 du contenu de la plage de données spécifiée
POW	Puissance exponentielle

---

## Macro LOG10

est disponible uniquement dans IBM Campaign. LOG10

### Syntaxe

LOG10(data)

### Paramètres

data

Valeurs numériques dont le logarithme base 10 doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

LOG10 calcule le logarithme base 10 des valeurs de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant le logarithme base 10 des nombres de la colonne d'entrée correspondante.

**Remarque :** Toutes les valeurs de la plage de données spécifiée doivent être supérieures à zéro. Dans le cas contraire, une cible vide est renvoyée pour chaque entrée non valide.

## Exemples

TEMP = LOG10(100)
Crée une colonne nommée TEMP contenant la valeur deux.
TEMP = LOG10(V1)
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au logarithme base 10 du contenu de la colonne V1.
TEMP = LOG10(V1:V3)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux logarithmes base 10 du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux logarithmes base 10 du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux logarithmes base 10 du contenu de la colonne V3.
TEMP = LOG10(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les logarithmes base 10 des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.
TEMP = LOG10(V1[1:5]:V2)
Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux logarithmes base 10 des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux logarithmes base 10 des lignes correspondantes de la colonne V2.

## Fonctions associées

Fonction	Description
LN ou LOG	Calcule le logarithme naturel du contenu de la plage de données spécifiée
LOG2	Calcule le logarithme base 2 du contenu de la plage de données spécifiée
POW	Puissance exponentielle

---

## Macro LOWER

est disponible dans IBM Campaign et IBM Interact. LOWER

### Syntaxe

LOWER(data)

### Paramètres

data

Valeur de chaîne à convertir en minuscules.

### Description

LOWER convertit chaque valeur de chaîne de la plage de données spécifiée en minuscules. Il renvoie une nouvelle colonne dont chaque cible contient la chaîne en minuscules de la cible source correspondante.



## Exemples

```
Temp = LOWER "GOLD"
```

Crée une colonne nommée Temp contenant "gold".

```
TEMP = LOWER( "JAN 15, 1997")
```

Crée une colonne nommée TEMP, qui contient la chaîne de texte ASCII " jan 15, 1997 ".

```
TEMP = LOWER( "Pressure")
```

Crée une colonne nommée TEMP, qui contient la chaîne de texte ASCII "pressure".

```
TEMP = LOWER(V1)
```

Crée une colonne nommée TEMP contenant les caractères en minuscules de chaque chaîne de la colonne V1.

---

## Macro LT

est disponible dans IBM Campaign et IBM Interact. LT

### Syntaxe

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

### Paramètres

data1

Plage de cibles numériques à comparer. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

data2

Nombres auxquels toutes les valeurs de la colonne spécifiée doivent être comparées. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

LT compare les deux plages de données spécifiées, puis renvoie un un si les valeurs du premier jeu de données sont inférieures à celles du deuxième jeu de données ou un zéro dans le cas contraire. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 comparée à la colonne correspondante de data2 (à savoir, la première colonne de data1 est comparée à la première colonne de data, la deuxième colonne à la deuxième colonne et ainsi de suite).

Si data2 est une constante, chaque valeur de data est comparée à cette valeur. Si data2 est une colonne, les calculs sont effectués ligne par ligne. Les valeurs de data1 sont comparées à la valeur de la première ligne de data2, la deuxième ligne

à la deuxième ligne et ainsi de suite. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** L'opérateur LT peut être abrégé en signe inférieur (<).

## Exemples

TEMP = 3 LT 4 ou TEMP = 3 < 4
Crée une colonne nommée TEMP contenant la valeur un (car trois est inférieur à quatre).
TEMP = V1 < 8
Crée une colonne nommée TEMP, dans laquelle chaque valeur est égale à un si la ligne correspondante de la colonne V1 est inférieure à huit ; sinon, la valeur est égale à zéro.
TEMP = V1:V3 < 2
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent au contenu de la colonne V1 comparé à la valeur deux, les valeurs de la colonne VX correspondent au contenu de la colonne V2 comparé à la valeur deux, et les valeurs de la colonne VY correspondent au contenu de la colonne V3 comparé à la valeur deux.
TEMP = V1 < V1
Crée une colonne nommée TEMP ne contenant que des zéros (car aucun nombre n'est inférieur à lui-même).
TEMP = V1 < V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 comparée à la valeur de ligne correspondante de la colonne V2.
TEMP = V1[10:20] < V2 ou TEMP = V1[10:20] < V2[1:11]
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les résultats de la comparaison des valeurs des lignes 10 à 20 de la colonne V1 aux lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
EQ	Renvoie TRUE si une plage de données est égale à une autre
GE	Renvoie TRUE si une plage de données est supérieure ou égale à une autre
GT	Renvoie TRUE si une plage de données est supérieure à une autre
LE	Renvoie TRUE si une plage de données est inférieure ou égale à une autre

---

## Macro LTRIM

est disponible dans IBM Campaign et IBM Interact. LTRIM

### Syntaxe

LTRIM(data)

## Paramètres

data

Chaîne de laquelle l'espace à gauche sera supprimé.

## Description

LTRIM supprime les premiers caractères d'espacement de chaque valeur de chaîne contenue dans la plage de données spécifiée, puis renvoie la chaîne convertie. Il renvoie une nouvelle colonne pour chaque colonne d'entrée.

## Exemples

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

Crée une chaîne nommée Temp qui contient "gold".

---

## Macro MAX

est disponible dans IBM Campaign et IBM Interact. MAX

## Syntaxe

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

## Paramètres

data

Valeurs numériques dont la valeur maximale doit être calculée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

keyword

Ce mot clé facultatif détermine la manière dont le calcul est effectué sur la plage de données d'entrée. Sélectionnez l'une des valeurs suivantes :

ALL - Effectue le calcul sur toutes les cibles de data (valeur par défaut)

COL - Effectue le calcul séparément pour chaque colonne de data

ROW - Effectue le calcul séparément pour chaque ligne de data

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

**Remarque :** De nombreuses fonctions macro acceptent les paramètres de mot clé {ALL | COL | ROW}. Ces mots clés ne s'appliquent pas dans IBM Campaign car les données d'entrées correspondent toujours à une colonne ou à une zone. La macro se comporte toujours comme si le mot clé COL a été spécifié. Par conséquent, vous n'avez pas besoin de spécifier ces mots clés lorsque vous utilisez IBM Campaign.

## Description

MAX calcule la valeur maximale des valeurs de la plage de données spécifiée. Il renvoie une nouvelle colonne unique contenant la valeur maximale.

## Exemples

TEMP = MAX(3) ou TEMP = MAX(3, ALL)
Crée une colonne nommée TEMP contenant la valeur trois.
TEMP = MAX(V1)
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la valeur maximale du contenu de la colonne V1.
TEMP = MAX(V1:V3)
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la valeur maximale des colonnes V1, V2 et V3.
TEMP = MAX(V1[10:20])
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la valeur maximale des cibles des lignes 10 à 20 de la colonne V1.
TEMP = MAX(V1[1:5]:V4)
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la valeur maximale des cibles des lignes 1 à 5 des colonnes V1 à V4.
TEMP = MAX(V1:V3, COL)
Crée trois colonnes nommées TEMP, VX et VY. La valeur unique de la colonne TEMP correspond à la valeur maximale du contenu de la colonne V1, la valeur unique de la colonne VX correspond à la valeur maximale du contenu de la colonne V2 et la valeur unique de la colonne VY correspond à la valeur maximale du contenu de la colonne V3.
TEMP = MAX(V1[1:5]:V3, COL)
Crée trois colonnes nommées TEMP, VX et VY contenant chacune une seule valeur. La valeur de la colonne TEMP correspond à la valeur maximale des cibles des lignes 1 à 5 de la colonne V1, la valeur de la colonne VX correspond à la valeur maximale des cibles des lignes 1 à 5 de la colonne V2, et la valeur de la colonne VY correspond à la valeur maximale des cibles des lignes 1 à 5 de la colonne V3.
TEMP = MAX(V1:V3, ROW)
Crée une colonne nommée TEMP dans laquelle chaque entrée de cible correspond à la valeur maximale de la ligne correspondante dans les colonnes V1, V2 et V3.
TEMP = MAX(V1[10:20]:V3, ROW)
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent la valeur maximale des valeurs des lignes 10 à 20 dans les colonnes V1 à V3. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
MIN	Calcule la valeur minimale d'une plage de cibles

---

## Macro MEAN

est disponible dans IBM Campaign et IBM Interact. MEAN

### Syntaxe

MEAN(data [, keyword])

### Paramètres

data

Valeurs numériques dont la moyenne arithmétique doit être calculée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

keyword

Ce mot clé facultatif détermine la manière dont le calcul est effectué sur la plage de données d'entrée. Sélectionnez l'une des valeurs suivantes :

ALL - Effectue le calcul sur toutes les cibles de data (valeur par défaut)

COL - Effectue le calcul séparément pour chaque colonne de data

ROW - Effectue le calcul séparément pour chaque ligne de data

Pour plus de détails sur l'utilisation des mots clé, voir «DATE», à la page 52.

**Remarque :** De nombreuses fonctions de macro utilisent les paramètres de mot clé {ALL | COL | ROW}. Ces mots clés ne s'appliquent pas dans IBM Campaign car les données d'entrées correspondent toujours à une colonne ou à une zone. La macro se comporte toujours comme si le mot clé COL a été spécifié. Par conséquent, vous n'avez pas besoin de spécifier ces mots clés lorsque vous utilisez IBM Campaign.

### Description

MEAN calcule la moyenne arithmétique des cibles de la plage de données spécifiée. Pour calculer la moyenne arithmétique, le contenu de l'ensemble des cibles est additionné, puis le résultat est divisé par le nombre de cible. Le nombre de colonnes renvoyé par MEAN dépend de keyword.

- Si keyword a la valeur ALL, MEAN renvoie une nouvelle colonne, contenant une valeur unique (la moyenne de toutes les cibles de data).
- Si keyword a la valeur COL, MEAN renvoie une nouvelle colonne pour chaque colonne d'entrée. Chaque nouvelle colonne contient une valeur (moyenne de toutes les cibles de la colonne d'entrée correspondante).
- Si keyword a la valeur ROW, MEAN renvoie une nouvelle colonne contenant la moyenne de chaque ligne de data.

**Remarque :** Les cibles vides sont exclues de la moyenne.

**Remarque :** MEAN est identique à la fonction macro AVG.

## Exemples

<code>TEMP = MEAN(V1)</code>
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la moyenne arithmétique du contenu de la colonne V1.
<code>TEMP = MEAN(V1:V3)</code>
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la moyenne arithmétique du contenu des colonnes V1, V2 et V3.
<code>TEMP = MEAN(V1[10:20])</code>
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la moyenne arithmétique des cibles des lignes 10 à 20 de la colonne V1.
<code>TEMP = MEAN(V1[1:5]:V4)</code>
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la moyenne arithmétique des cibles des lignes 1 à 5 des colonnes V1 à V4.
<code>TEMP = MEAN(V1:V3, COL)</code>
Crée trois colonnes nommées TEMP, VX et VY. La valeur unique de la colonne TEMP correspond à la moyenne arithmétique du contenu de la colonne V1, la valeur unique de la colonne VX correspond à la moyenne arithmétique du contenu de la colonne V2 et la valeur unique de la colonne VY correspond à la moyenne arithmétique du contenu de la colonne V3.
<code>TEMP = MEAN(V1[10:20]:V3, COL)</code>
Crée trois colonnes nommées TEMP, VX et VY contenant chacune une seule valeur. La valeur de la colonne TEMP correspond à la moyenne arithmétique des cibles des lignes 10 à 20 de la colonne V1, la valeur de la colonne VX correspond à la moyenne arithmétique des cibles des lignes 10 à 20 de la colonne V2, et la valeur de la colonne VY correspond à la moyenne arithmétique des cibles des lignes 10 à 20 de la colonne V3.
<code>TEMP = MEAN(V1:V3, ROW)</code>
Crée une colonne nommée TEMP dans laquelle chaque entrée de cible correspond à la moyenne arithmétique de la ligne correspondante dans les colonnes V1, V2 et V3.
<code>TEMP = MEAN(V1[1:5]:V3,ROW)</code>
Crée une colonne nommée TEMP, dans laquelle les cibles des lignes 1 à 5 contiennent la moyenne arithmétique de la ligne correspondante dans les colonnes V1 à V3. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
SUM ou TOTAL	Calcule la somme d'une plage de cibles

---

## Macro MIN

La macro MIN est disponible dans IBM Campaign et IBM Interact.

### Syntaxe

`MIN(data [, keyword])`

## Paramètres

data

Valeurs numériques dont la valeur minimale doit être calculée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

keyword

Ce mot clé facultatif détermine la manière dont le calcul est effectué sur la plage de données d'entrée. Sélectionnez l'une des valeurs suivantes :

ALL - Effectue le calcul sur toutes les cibles de data (valeur par défaut)

COL - Effectue le calcul séparément pour chaque colonne de data

ROW - Effectue le calcul séparément pour chaque ligne de data

Pour plus de détails sur l'utilisation des mots clé, voir «DATE», à la page 52.

**Remarque :** De nombreuses fonctions de macro utilisent les paramètres de mot clé {ALL | COL | ROW}. Ces mots clés ne s'appliquent pas dans **IBM Campaign** car les données d'entrées correspondent toujours à une colonne ou à une zone. La macro se comporte toujours comme si le mot clé COL a été spécifié. Par conséquent, vous n'avez pas besoin de spécifier ces mots clés lorsque vous utilisez **IBM Campaign**.

## Description

MIN calcule la valeur minimale de toutes les cibles de la plage de données spécifiée. Il renvoie une colonne unique contenant la valeur minimale.

## Exemples

TEMP = MIN(V1)

Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la valeur minimale de la colonne V1.

TEMP = MIN(V1:V3)

Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la valeur minimale des colonnes V1, V2 et V3.

TEMP = MIN(V1[10:20])

Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la valeur minimale des cibles des lignes 10 à 20 de la colonne V1.

TEMP = MIN(V1[1:5]:V4)

Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la valeur minimale des cibles des lignes 1 à 5 des colonnes V1 à V4.

TEMP = MIN(V1:V3, COL)
Crée trois colonnes nommées TEMP, VX et VY. La valeur unique de la colonne TEMP correspond à la valeur minimale de la colonne V1, la valeur unique de la colonne VX correspond à la valeur minimale de la colonne V2 et la valeur unique de la colonne VY correspond à la valeur minimale de la colonne V3.
TEMP = MIN(V1[1:5]:V3, COL)
Crée trois colonnes nommées TEMP, VX et VY contenant chacune une seule valeur. La valeur de la colonne TEMP correspond à la valeur minimale des cibles des lignes 1 à 5 de la colonne V1, la valeur de la colonne VX correspond à la valeur minimale des cibles des lignes 1 à 5 de la colonne V2, et la valeur de la colonne VY correspond à la valeur minimale des cibles des lignes 1 à 5 de la colonne V3.
TEMP = MIN(V1:V3, ROW)
Crée une colonne nommée TEMP dans laquelle chaque entrée de cible correspond à la valeur minimale de la ligne correspondante dans les colonnes V1, V2 et V3.
TEMP = MIN(V1[10:20]:V3, ROW)
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent la valeur minimale des valeurs des lignes 1 à 5 dans les colonnes V1 à V3. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
MAX	Calcule la valeur maximale d'une plage de cibles
MAX_TO_INDEX	Renvoie l'index de colonne de la valeur maximale pour chaque ligne de la colonne spécifiée

---

## Macro MINUS

est disponible dans IBM Campaign et IBM Interact. MINUS

### Syntaxe

data MINUS subtrahend data - subtrahend

### Paramètres

data

Plage de cibles contenant les nombres desquels une valeur doit être soustraite. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

subtrahend

Nombres à soustraire de toutes les valeurs de la colonne spécifiée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans subtrahend



doit correspondre au nombre de colonnes dans data, à moins que subtrahend ne soit une constante. Pour la définition du format de subtrahend (identique à data), voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

MINUS soustrait subtrahend de la plage de données spécifiée data. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de subtrahend (à savoir, la première colonne de data soustrait la première colonne de subtrahend, la deuxième colonne soustrait la deuxième colonne et ainsi de suite).

Si subtrahend est une constante, cette valeur est soustraite de chaque valeur de data. Si subtrahend contient une ou plusieurs colonnes, les calculs sont effectués ligne par ligne entre une colonne de data et une colonne de subtrahend. La première ligne de data soustrait la valeur de la première ligne de subtrahend, la seconde ligne soustrait la valeur de la deuxième ligne, etc. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** L'opérateur MINUS peut être abrégé en signe moins ou tiret (-).

## Exemples

<p>TEMP = 7 MINUS 4 ou TEMP = 7 - 4</p> <p>Crée une colonne nommée TEMP contenant la valeur trois.</p>
<p>TEMP = V1 - 8</p> <p>Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au contenu de la colonne V1 moins huit.</p>
<p>TEMP = V1:V3 - 2</p> <p>Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent au contenu de la colonne V1 moins deux, les valeurs de la colonne VX correspondent au contenu de la colonne V2 moins deux et les valeurs de la colonne VY correspondent au contenu de la colonne V3 moins deux.</p>
<p>TEMP = V1 - V1</p> <p>Crée une colonne nommée TEMP ne contenant que des zéros (car tout nombre moins lui-même est égal à zéro).</p>
<p>TEMP = V1 - V2</p> <p>Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 moins la valeur de ligne correspondante de la colonne V2.</p>
<p>TEMP = V1:V3 -V4:V6</p> <p>Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 moins les valeurs de ligne correspondantes de la colonne V4. La colonne VX soustrait la colonne V5 de V2. La colonne VY soustrait la colonne V6 de V3.</p>
<p>TEMP = V1[10:20] - V2 ou TEMP = V1[10:20] - V2[1:11]</p> <p>Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les valeurs des lignes 10 à 20 de la colonne V1 moins les valeurs des lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.</p>

## Fonctions associées

Fonction	Description
PLUS	Ajoute le contenu de deux plages de données
SUM ou TOTAL	Calcule la somme d'une plage de cibles

---

## Macro MOD

est disponible dans IBM Campaign et IBM Interact. MOD

### Syntaxe

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

### Paramètres

data

Valeurs entières dont le modulo doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

divisor

Entier de base non nulle par rapport auquel le modulo doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans divisor doit correspondre au nombre de colonnes dans data, à moins que divisor ne soit une constante. Pour la définition du format de divisor (identique à data), voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

MOD calcule le reste de la division de la plage de données spécifiée par une valeur spécifiée. Ce reste est calculé en divisant par divisor chaque valeur et en renvoyant le reste. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant les nombres de data modulo divisor. Le reste possède le même signe (positif ou négatif) que data.

Si divisor est une constante, chaque valeur de la colonne spécifiée est calculée modulo cette valeur. Si divisor est une colonne, les calculs sont effectués ligne par ligne. Les valeurs de data sont calculées modulo la valeur de la première ligne de divisor et le même calcul est effectué entre les deuxièmes lignes, et ainsi de suite. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** Si divisor est égal à zéro, une erreur de division par zéro est renvoyée.

**Remarque :** L'opérateur MOD peut être abrégé en signe pourcentage (%). Par exemple, TEMP = 5 % 3 est équivalent à TEMP = 5 MOD 3.

## Exemples

TEMP = 10 MOD 8 ou TEMP = 10 % 8
Crée une colonne nommée TEMP contenant la valeur 2.
TEMP = -10 % 8
Crée une colonne nommée TEMP contenant la valeur -2.
TEMP = V1 % 8
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au contenu de la colonne V1 modulo huit.
TEMP = V1:V3 % 2
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux valeurs modulo deux du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux valeurs modulo deux du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux valeurs modulo deux du contenu de la colonne V3.
TEMP = V1 % V1
Crée une colonne nommée TEMP contenant un zéro pour chaque entrée de la colonne V1. En effet, tout nombre modulo lui-même renvoie la valeur zéro.
TEMP = V1 % V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 modulo la valeur de ligne correspondante de la colonne V2. Notez que si V2=V1, des zéros sont renvoyés, comme dans l'exemple précédent.
TEMP = V1:V3 % V4:V6
Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 modulo les valeurs de ligne correspondantes de la colonne V4. La colonne VX contient les résultats de la colonne V2 modulo V5. La colonne VY contient les résultats de la colonne V3 modulo V6.
TEMP = V1[10:20] % V2 ou TEMP = V1[10:20] % V2[1:11]
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles correspondent aux valeurs des lignes 10 à 20 de la colonne V1 modulo les valeurs des lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
DIV	Divise une plage de données spécifiée par une autre
MOD	Calcule le modulo du contenu de la plage de données spécifiée

## Macro MONTHOF

est disponible uniquement dans IBM Campaign. MONTHOF

### Syntaxe

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

## Paramètres

date\_string

Texte représentant une date valide.

input\_format

Un des mots clés du tableau ci-après, spécifiant le format de date de date\_string.

## Description

MONTHOF renvoie le mois sous la forme d'un nombre pour la date indiquée par date\_string. Si input\_format n'est pas fourni, le mot clé par défaut DELIM\_M\_D\_Y est utilisé.

## Exemples

MONTHOF("012171",MMDDYY) renvoie le nombre 1.

**Remarque :** Pour des informations supplémentaires sur les formats de date valides, voir «DATE», à la page 52.

## Fonctions associées

Fonction	Description
DAYOF	Renvoie le jour de la semaine sous la forme d'un nombre.
WEEKDAYOF	Renvoie le jour de la semaine de la semaine sous forme de nombre.
YEAROF	Renvoie l'année sous la forme d'un nombre.

---

## Macro MULT

est disponible dans IBM Campaign et IBM Interact. MULT

## Syntaxe

data MULT multiplier data \* multiplier

## Paramètres

data

Valeurs numériques à multiplier. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

multiplier

Nombre par lequel toutes les valeurs de la colonne spécifiée doivent être multipliées. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de

cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans multiplier doit correspondre au nombre de colonnes dans data, à moins que multiplier ne soit une constante. Pour la définition du format de multiplier (identique à data), voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

MULT multiplie les valeurs des deux plages de données spécifiées. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant les nombres de data multipliés par multiplier. Si multiplier est une constante, chaque valeur de data est multipliée par cette valeur. Si multiplier est une colonne, les calculs sont effectués ligne par ligne. Les valeurs de data sont multipliées par la valeur de la première ligne de multiplier, la seconde ligne, par la seconde ligne, etc. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** L'utilisation d'une colonne contenant le même nombre  $x$  dans chaque ligne que multiplier revient à utiliser la constante  $x$  pour multiplier.

**Remarque :** L'opérateur MULT peut être abrégé en astérisque (\*).

## Exemples

<p>TEMP = 8 MULT 4 ou TEMP = 8 * 4</p> <p>Crée une colonne nommée TEMP contenant la valeur 32.</p>
<p>TEMP = V1 * 8</p> <p>Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au contenu de la colonne V1 multiplié par huit.</p>
<p>TEMP = V1:V3 * 2</p> <p>Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent à deux fois le contenu de la colonne V1, les valeurs de la colonne VX correspondent à deux fois le contenu de la colonne V2 et les valeurs de la colonne VY correspondent à deux fois le contenu de la colonne V3.</p>
<p>TEMP = V1 * V1</p> <p>Crée une colonne nommée TEMP contenant le carré de chaque valeur de la colonne V1.</p>
<p>TEMP = V1 * V2</p> <p>Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 multipliée par la valeur de ligne correspondante de la colonne V2.</p>
<p>TEMP = V1:V3 * V4:V6</p> <p>Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 multipliées par les valeurs de ligne correspondantes de la colonne V4. La colonne VX multiplie la colonne V2 par V5. La colonne VY multiplie la colonne V3 par V6.</p>
<p>TEMP = V1[10:20] * V2 ou TEMP = V1[10:20] * V2[1:11]</p> <p>Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les valeurs des lignes 10 à 20 de la colonne V1 fois les valeurs des lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.</p>

## Fonctions associées

Fonction	Description
DIV	Divise une plage de données spécifiée par une autre
EXP	Calcule le nombre naturel (e) élevé à la puissance du contenu de chaque cible de la plage de données spécifiée
POW	Calcule une valeur de base élevée aux puissances exponentielles spécifiées

---

## Macro NE

est disponible dans IBM Campaign et IBM Interact. NE

### Syntaxe

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

### Paramètres

data1

Plage de cibles à comparer. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

data2

Nombres auxquels toutes les valeurs de la colonne spécifiée doivent être comparées. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans data2 doit correspondre au nombre de colonnes dans data1, à moins que data2 ne soit une constante. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

NE compare les deux plages de données spécifiées, puis renvoie un un si les valeurs ne sont pas égales ou un zéro si elles le sont. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 comparée à la colonne correspondante de data2 (à savoir, la première colonne de data1 est comparée à la première colonne de data, la deuxième colonne à la deuxième colonne et ainsi de suite).

Si data2 est une constante, chaque valeur de data1 est comparée à cette valeur. Si data2 est une colonne, les calculs sont effectués ligne par ligne. Les valeurs de la première ligne de data1 sont comparées à la valeur de la première ligne de data2, la deuxième ligne à la deuxième ligne et ainsi de suite. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** L'utilisation d'une colonne contenant le même nombre  $x$  dans chaque ligne que data2 revient à utiliser la constante  $x$  pour data2.

**Remarque :** L'opérateur NE peut être abrégé en point d'exclamation suivi d'un signe égal (!=) ou en signe inférieur suivi d'un signe supérieur (<=).

## Exemples

TEMP = 3 NE 4 ou TEMP = 3 != 4 TEMP = 3 <= 4
Crée une colonne nommée TEMP contenant la valeur un (car trois n'est pas égal à quatre).
TEMP = V1 != 8
Crée une colonne nommée TEMP, dans laquelle chaque valeur est égale à un si la ligne correspondante de la colonne V1 n'est pas égale à huit ; sinon, la valeur est égale à zéro.
TEMP = V1:V3 != 2
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent au contenu de la colonne V1 comparé à la valeur deux, les valeurs de la colonne VX correspondent au contenu de la colonne V2 comparé à la valeur deux, et les valeurs de la colonne VY correspondent au contenu de la colonne V3 comparé à la valeur deux.
TEMP = V1 != V1
Crée une colonne nommée TEMP ne contenant que des zéros (car tout nombre est égal à lui-même).
TEMP = V1 != V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 comparée à la valeur de ligne correspondante de la colonne V2.
TEMP = V1:V3 != V4:V6
Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 comparées aux valeurs de ligne correspondantes de la colonne V4. La colonne VX compare les colonnes V2 et V5. La colonne VY compare les colonnes V3 et V6.
TEMP = V1[10:20] != V2 ou TEMP = V1[10:20] != V2[1:11]
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les résultats de la comparaison des valeurs des lignes 10 à 20 de la colonne V1 aux lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
EQ	Renvoie TRUE si une plage de données est égale à une autre
GE	Renvoie TRUE si une plage de données est supérieure ou égale à une autre
GT	Renvoie TRUE si une plage de données est supérieure à une autre
LE	Renvoie TRUE si une plage de données est inférieure ou égale à une autre
LT	Renvoie TRUE si une plage de données est inférieure à une autre

---

## Macro NOT

est disponible dans IBM Campaign et IBM Interact. NOT

### Syntaxe

NOT(data) ! data

### Paramètres

data

Valeurs numériques dont le NOT logique doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

NOT renvoie le NOT logique des valeurs de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant le NOT logique des valeurs de la colonne d'entrée correspondante. Cette fonction renvoie la valeur zéro pour les valeurs non nulles et la valeur un pour les valeurs nulles.

**Remarque :** L'opérateur NOT peut être abrégé en point d'exclamation (!). Utilisez le point d'exclamation avant la valeur (par exemple, pour spécifier NOT(V1), vous pouvez simplement saisir !V1).

### Exemples

TEMP = NOT(3.2) ou TEMP = !1 Crée une colonne nommée TEMP contenant la valeur zéro.
TEMP = !0 ou TEMP = !(2+2=3) Crée une colonne nommée TEMP contenant la valeur un.
TEMP = !V1 Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au NOT logique des valeurs de la colonne V1.
TEMP = !V1:V3 Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux valeurs NOT logiques des valeurs de la colonne V1, les valeurs de la colonne VX correspondent aux valeurs NOT logiques des valeurs de la colonne V2 et les valeurs de la colonne VY correspondent aux valeurs NOT logiques des valeurs de la colonne V3.
TEMP = !V1[10:20] Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les valeurs NOT logiques des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.



```
TEMP = !V1[1:5]:V2
```

Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux valeurs NOT logiques des valeurs des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux valeurs NOT logiques des valeurs des lignes correspondantes de la colonne V2.

## Fonctions associées

Fonction	Description
AND	Calcule le et logique (AND) entre deux plages de données spécifiées
INVERSE	Calcule la valeur négative du contenu de la plage de données spécifiée
OR	Calcule le OR logique entre deux plages de données spécifiées
SIGN	Détermine le signe (positif ou négatif) des valeurs dans la plage de données spécifiée

---

## Macro NUMBER

La macro NUMBER est disponible dans IBM Campaign et IBM Interact.

### Syntaxe

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

### Paramètres

data

Données de texte ASCII à convertir en valeurs numériques. Il peut s'agir de texte ASCII entre guillemets, d'une colonne de texte, d'une plage de cibles contenant du texte ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

conversion\_keyword

Ce mot clé facultatif indique comment interpréter les formats de texte des dates et des heures. Sélectionnez l'un des mots clés du tableau ci-après.

**Remarque :** Si ce paramètre n'est pas spécifié, la valeur par défaut est 1.

Mot clé de conversion	Format	Description
Classe 0	#####	Convertit les 5 premiers caractères de chaque chaîne de texte en un numéro unique
1	\$ (valeur par défaut)	Convertit les valeurs en dollars en valeurs numériques (par exemple, "\$123.45" en 123.45)

<b>Mot clé de conversion</b>	<b>Format</b>	<b>Description</b>
2	%	Convertit un pourcentage en valeur numérique (par exemple, "50%" en 0,5)
3	mm/dd/yy hh:mm	Convertit une date et une heure en nombre de jours écoulés depuis le 1 janvier 0000 (1900 est automatiquement ajouté à l'année yy)
4	dd- <i>mmm</i> -yy	Convertit une date en nombre de jours écoulés depuis le 1 janvier 0000 (1900 est automatiquement ajouté à l'année yy)
5	mm/dd/yy	Convertit une date en nombre de jours écoulés depuis le 1 janvier 0000 (1900 est automatiquement ajouté à l'année yy)
6	<i>mmm</i> -yy	Convertit une date en nombre de jours écoulés entre le premier jour du mois spécifié et le 1er janvier 0000 (1900 est automatiquement ajouté à l'année yy)
7	dd- <i>mmm</i>	Convertit une date en nombre de jours écoulés depuis le début de l'année (par exemple, "01-FEB" en 32)
8	<i>mmm</i>	Convertit une abréviation de mois en trois lettres en une valeur comprise entre 1 et 12 (par exemple, "DEC" en 12)
9	{January   February   March ... }	Convertit un nom de mois en toutes lettres en valeur comprise entre 1 et 12 (par exemple, "March" en 3)
10	{Sun   Mon   Tue ... }	Convertit une abréviation de jour de la semaine sur trois caractères en valeur comprise entre 0 et 6, dimanche correspondant au début de la semaine (par exemple, "Sun" en 0)
11	{Sunday   Monday   Tuesday ... }	Convertit un nom de jour de la semaine en toutes lettres en valeur comprise entre 0 et 6, dimanche correspondant au début de la semaine (par exemple, "Monday" en 1)
12	hh:mm:ss {AM   PM}	Convertit l'heure en nombre de secondes écoulées depuis 00:00:00 AM (minuit) (par exemple, "01:00:00 AM" en 3600)
13	hh:mm:ss	Convertit l'heure en nombre de secondes écoulées depuis 00:00:00 AM (minuit) (par exemple, "01:00:00" en 3600)
14	hh:mm {AM   PM}	Convertit l'heure en nombre de minutes écoulées depuis 00:00:00 AM (minuit) (par exemple, "01:00 AM" en 60)
15	hh:mm	Convertit l'heure en nombre de minutes écoulées depuis 00:00:00 AM (minuit) (par exemple, "01:00" en 60)
16	mm:ss	Convertit l'heure en nombre de secondes écoulées depuis 00:00:00 AM (minuit) (par exemple, "30:00" en 1800)

<b>Mot clé de conversion</b>	<b>Format</b>	<b>Description</b>
17	ddmm	Convertit une date en nombre de jours écoulés depuis le début de l'année (par exemple, "3101" en 31)
18	ddmmm	Convertit une date en nombre de jours écoulés depuis le début de l'année (par exemple, "31JAN" en 31)
19	ddmmyy	Convertit une date en nombre de jours écoulés depuis le 1 janvier 0000 (1900 est automatiquement ajouté à l'année si yy est inférieur ou égal à 20 ; sinon, 2000 est ajouté)
20	ddmmyyyy	Convertit une date en nombre de jours écoulés depuis le 1er janvier 0000 (par exemple, "31JAN0000" en 31)
21	ddmmyy	Convertit une date en nombre de jours écoulés depuis le 1 janvier 0000 (1900 est automatiquement ajouté à l'année si yy est inférieur ou égal à 20 ; sinon, 2000 est ajouté)
22	ddmmyyyy	Convertit une date en nombre de jours écoulés depuis le 1er janvier 0000 (par exemple, "31010000" en 31)
23	mmdd	Convertit une date en nombre de jours écoulés depuis le début de l'année (par exemple, "0131" en 31)
24	mmddy	Convertit une date en nombre de jours écoulés depuis le 1 janvier 0000 (1900 est automatiquement ajouté à l'année si yy est inférieur ou égal à 20 ; sinon, 2000 est ajouté)
25	mmddyyyy	Convertit une date en nombre de jours écoulés depuis le 1er janvier 0000 (par exemple, "01010001" en 366)
26	mmm	Convertit une abréviation de mois en trois lettres en une valeur comprise entre 1 et 12 (par exemple, "MAR" en 3) [Remarque : identique au mot clé de conversion 8]
27	mmdd	Convertit une date en nombre de jours écoulés depuis le début de l'année (par exemple, "JAN31" en 31)
28	mmddy	Convertit une date en nombre de jours écoulés depuis le 1 janvier 0000 (1900 est automatiquement ajouté à l'année si yy est inférieur ou égal à 20 ; sinon, 2000 est ajouté)
29	mmddyyyy	Convertit une date en nombre de jours écoulés depuis le 1er janvier 0000 (par exemple, "FEB010001" en 32)

<b>Mot clé de conversion</b>	<b>Format</b>	<b>Description</b>
30	mmmyy	Convertit une date en nombre de jours écoulés entre le 1er du mois spécifié et le 1er janvier 0000 (1900 est automatiquement ajouté à l'année si yy est inférieur ou égal à 20 ; sinon, 2000 est ajouté)
31	mmmyyyy	Convertit une date en nombre de jours écoulés entre le 1er du mois spécifié et le 1er janvier 0000 (par exemple, "FEB0001" en 32)
32	mmyy	Convertit une date en nombre de jours écoulés entre le 1er du mois spécifié et le 1er janvier 0000 (1900 est automatiquement ajouté à l'année si yy est inférieur ou égal à 20 ; sinon, 2000 est ajouté)
33	mmyyyy	Convertit une date en nombre de jours écoulés entre le 1er du mois spécifié et le 1er janvier 0000 (par exemple, "020001" en 32)
34	yymm	Convertit une date en nombre de jours écoulés entre le 1er du mois spécifié et le 1er janvier 0000 (1900 est automatiquement ajouté à l'année si yy est inférieur ou égal à 20 ; sinon, 2000 est ajouté)
35	yymdd	Convertit une date en nombre de jours écoulés depuis le 1 janvier 0000 (1900 est automatiquement ajouté à l'année si yy est inférieur ou égal à 20 ; sinon, 2000 est ajouté)
36	yymm	Convertit une date en nombre de jours écoulés entre le 1er du mois spécifié et le 1er janvier 0000 (1900 est automatiquement ajouté à l'année si yy est inférieur ou égal à 20 ; sinon, 2000 est ajouté)
37	yymmdd	Convertit une date en nombre de jours écoulés depuis le 1 janvier 0000 (1900 est automatiquement ajouté à l'année si yy est inférieur ou égal à 20 ; sinon, 2000 est ajouté)
38	yyyy	Convertit le nombre d'années écoulées depuis l'an 0000 (par exemple, "1998" en 1998)
39	yyymm	Convertit une date en nombre de jours écoulés entre le 1er du mois spécifié et le 1er janvier 0000 (par exemple, "000102" en 32)
40	yyymmdd	Convertit une date en nombre de jours écoulés depuis le 1er janvier 0000 (par exemple, "00010201" en 32)
41	yyymmm	Convertit une date en nombre de jours écoulés entre le 1er du mois spécifié et le 1er janvier 0000 (par exemple, "000102" en 32)
42	yyymmmdd	Convertit une date en nombre de jours écoulés depuis le 1er janvier 0000 (par exemple, "0001FEB01" en 32)

<b>Mot clé de conversion</b>	<b>Format</b>	<b>Description</b>
43	<day>* <month>	Convertit toute date délimitée avec le jour suivi du mois en nombre de jours écoulés depuis le début de l'année (par exemple, "15-JAN" en 15)
44	<day>* <month>* <year>	Convertit toute date délimitée avec le jour précédant le mois et suivi de l'année en nombre de jours écoulés depuis le 1er janvier 0000 (par exemple, "1/1/0001" en 366)
45	<month>* <day>	Convertit toute date délimitée avec le mois suivi du jour en nombre de jours écoulés depuis le début de l'année (par exemple, "JAN 31" en 31)
46	<month>* <day>* <year>	Convertit toute date délimitée avec le mois suivi du jour, suivi de l'année en nombre de jours écoulés depuis le 1er janvier 0000 (par exemple, "JAN 1, 0001" en 366)
47	<month>* <year>	Convertit toute date délimitée avec le mois suivi de l'année en nombre de jours écoulés entre le premier jour du mois spécifié et le 1er janvier 0000
48	<year>* <month>	Convertit toute date délimitée avec l'année suivi du mois en nombre de jours écoulés entre le premier jour du mois spécifié et le 1er janvier 0000
49	<year>* <month>* <day>	Convertit toute date délimitée avec le mois suivi du jour, suivi de l'année en nombre de jours écoulés depuis le 1er janvier 0000 (par exemple, "0001/01/01" en 366)
50	yy	Convertit l'année en nombre d'années écoulées depuis l'an 0000 (par exemple, "97" en 97)
51	mm	Convertit le mois en valeur comprise entre 1 et 12 (par exemple, "SEP" en 9)
52	dd	Convertit le jour en valeur comprise entre 1 et 31 (par exemple, "28" en 28)
53	{January   February   March ... }	Convertit un nom de mois en toutes lettres en valeur comprise entre 1 et 12 (par exemple, "March" en 3) [Remarque : identique au mot clé de conversion 9]
54	{Sunday   Monday   Tuesday ... }	Convertit un nom de jour de la semaine en toutes lettres en valeur comprise entre 1 et 7, dimanche correspondant au début de la semaine (par exemple, "Sunday" en 1)
55	{Sun   Mon   Tue ... }	Convertit une abréviation de jour de la semaine sur trois caractères en valeur comprise entre 1 et 7, dimanche correspondant au début de la semaine (par exemple, "Sun" en 1)

## Description

NUMBER convertit les valeurs de texte dans la plage de données spécifiée en valeurs numériques, à l'aide du format spécifié pour convertir les dates et les heures. Si une chaîne de texte ne peut pas être analysée à l'aide du mot clé `conversion_keyword` spécifié, NUMBER génère une erreur. Le format 0 convertit les cinq premiers caractères de chaque chaîne de texte en nombre différent pour chaque chaîne de texte unique. C'est un moyen simple de changer une colonne de texte en classes uniques pour les sorties dans un discriminant.

Les formats délimités (mots clés de conversion 43 à 49) acceptent les caractères suivants comme délimiteurs :

- / (barre oblique)
- - (tiret)
- , (virgule)
- " " (espace)
- : (deux-points)

Les mois peuvent être représentés sous la forme `mm` ou `mmm` ; les jours peuvent être représentés sous la forme `d` ou `dd` ; les années peuvent être représentées sous la forme `yy` ou `yyyy`.

**Remarque :** Dans le cadre de la conformité à l'an 2000, toutes les années des dates peuvent être désignées sous la forme `yyyy` au lieu de `yy`. Pour la compatibilité amont, dans les mots clés de conversion 1 à 16, 1900 est automatiquement ajouté à `yy` (années sur deux chiffres). Pour les mots clés de conversion 17 à 55, 2000 est automatiquement ajouté à `yy < threshold` ; 1900 est automatiquement ajouté à `yy ≥ threshold`.

**Remarque :** La valeur `threshold` de l'an 2000 est définie dans l'onglet **Nettoyage des données** de la fenêtre **Paramètres avancés** (appelez-la en sélectionnant **Options > Paramètres > Paramètres avancés**).

**Remarque :** Si vous changez la valeur de seuil an 2000, vous devez mettre à jour toutes les fonctions macro à l'aide de la fonction macro NUMBER pour manipuler les valeurs de date avec des années sur 2 chiffres. Pour forcer une mise à jour d'une fonction macro, vous pouvez effectuer toute changement (par exemple, ajouter un espace et le supprimer) et cliquer sur l'icône en forme de coche pour accepter la changement.

**Remarque :** Lorsque vous utilisez le format 0, seuls les cinq premiers caractères de chaque chaîne de texte sont utilisés pour générer un nombre unique. Toutes les chaînes contenant ces cinq premiers caractères seront converties dans la même valeur numérique. La même chaîne de texte générera la même valeur numérique à chaque fois, même entre des feuilles de calcul différentes. Si nécessaire, utilisez des macros de chaîne pour manipuler les chaînes de sorte que les cinq premiers caractères définissent une classe de manière unique. Notez que les valeurs numériques résultantes peuvent être très petites. Utilisez la fenêtre **Formats d'affichage** pour augmenter le nombre de décimales à afficher ou changer le format en mode exponentiel (`00E+00`).

## Exemples

<pre>TEMP = NUMBER("\$1.23") ou TEMP = NUMBER("123%", 2)</pre>
Crée une colonne nommée TEMP contenant le nombre 1.23.
<pre>TEMP = NUMBER(column("Jan", "Mar", "Dec", 8))</pre>
Crée une colonne nommée TEMP contenant les nombres 1, 3 et 12.
<pre>TEMP = NUMBER("1:52 PM", 14)</pre>
Crée une colonne nommée TEMP contenant le nombre 832.
<pre>TEMP = NUMBER("1/1/95", 5)</pre>
Crée une colonne nommée TEMP contenant le nombre 728660.
<pre>TEMP = NUMBER(V1)</pre>
Crée une colonne nommée TEMP contenant les valeurs numériques des chaînes de texte de la colonne V1. Les valeurs en dollars sont correctement converties en valeurs numériques. ??? 'est renvoyé pour les chaînes de texte qui ne peuvent pas être analysées à l'aide du format \$.
<pre>TEMP = NUMBER(V1:V3, 4)</pre>
Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs numériques des chaînes de texte de la colonne V1. La colonne VX contient les valeurs numériques des chaînes de texte de la colonne V2. La colonne VY contient les valeurs numériques des chaînes de texte de la colonne V3. Les dates au format dd-mmm-yy sont converties en nombre de jours de décalage à partir du 1er janvier 0000. ??? 'sont renvoyées pour les chaînes de texte qui ne peuvent pas être analysées à l'aide du format \$.
<pre>TEMP = NUMBER(V1[10:20]:V2, 10)</pre>
Crée deux colonnes nommées TEMP et VX. La colonne TEMP contient les valeurs numériques des chaînes de texte des lignes 10 à 20 de la colonne V1. La colonne VX contient les valeurs numériques des chaînes de texte des lignes 10 à 20 de la colonne V2. Toutes les représentations standard de jour de la semaine à trois caractères sont converties en nombres de 0 à 6 (0 = dimanche, 6 = samedi). En l'absence de correspondance pour un nom de jour de la semaine, ??? est renvoyé.
<pre>TEMP = NUMBER(V1, 0)</pre>
Si la colonne V1 ne contient que des chaînes de texte à cinq chiffres, crée une colonne nommée TEMP contenant une valeur numérique différente par chaîne unique.

## Fonctions associées

Fonction	Description
WEEKDAY	Convertit les chaînes de date textuelles ASCII en jour de la semaine

---

## Macro OR

est disponible dans IBM Campaign et IBM Interact. OR

### Syntaxe

data1 OR data2 data1 || data2

## Paramètres

data1

Nombres dont le OR logique doit être calculé avec les valeurs de data2. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

data2

Nombres dont le OR logique doit être calculé avec les valeurs de data1. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans data2 doit correspondre au nombre de colonnes dans data1, à moins que data2 ne soit une constante. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

OR calcule le OR logique entre les deux plages de données spécifiées. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 dont le OR logique avec la colonne correspondante de data2 a été calculé (à savoir, le OR logique de la première colonne de data1 avec la première colonne de data, celui de la deuxième colonne avec la deuxième colonne et ainsi de suite).

Si data2 est une constante, le OR logique de chaque valeur de data1 avec cette valeur est calculé. Si data2 contient une ou plusieurs colonnes, les calculs sont effectués ligne par ligne entre une colonne de data1 et une colonne de data2. Le OR logique de la première ligne de data1 avec la première ligne de data2 est calculé, la seconde ligne avec la seconde ligne, etc. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** L'utilisation d'une colonne contenant le même nombre  $x$  dans chaque ligne que data2 revient à utiliser la constante  $x$  pour data2.

**Remarque :** L'opérateur OR peut être abrégé en double barre verticale (||). Utilisez la double barre verticale pour séparer les deux arguments (par exemple, pour spécifier V1 OR 3, vous pouvez simplement entrer V1 || 3).

## Exemples

```
TEMP = 1 OR 8 ou TEMP = 1 || 8
```

Crée une colonne nommée TEMP contenant la valeur un (tout nombre non nul prend la valeur un).

```
TEMP = V1 || 1
```

Crée une colonne nommée TEMP ne contenant que des uns (un OR entre toute valeur et le nombre un génère un).



TEMP = V1    V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 dont le OR logique est calculé avec la valeur de ligne correspondante de la colonne V2.
TEMP = V1:V3    V4:V6
Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 dont le OR logique avec les valeurs de ligne correspondantes de la colonne V4 a été calculé. La colonne VX contient les valeurs résultant du OR logique des colonnes V2 et V5. La colonne VY contient les valeurs résultant du OR logique des colonnes V3 et V6.
TEMP = V1[10:20]    V2
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent le résultat du OR logique entre les valeurs des lignes 10 à 20 des colonnes V1 et V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
AND	Calcule le et logique (AND) entre deux plages de données spécifiées
NOT	Calcule le non logique (NON) du contenu de la plage de données spécifiée

---

## Macro POSITION

est disponible uniquement dans IBM Campaign. POSITION

### Syntaxe

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

### Paramètres

colName

Valeur d'une colonne (doit être de type string).

pattern

Modèle ou chaîne que vous recherchez.

start

Octet à partir duquel la recherche commence.

occurrence

Indiquez une valeur pour n, où vous recherchez la n ème occurrence du modèle à renvoyer.

## Description

POSITION renvoie la position de l'octet de départ du modèle ou de la chaîne, dans la valeur d'une colonne (colName) qui doit être de type chaîne. Si vous donnez une valeur au paramètre début, la recherche commence à ce point. L'occurrence correspond à la n<sup>ème</sup> occurrence du modèle à renvoyer.

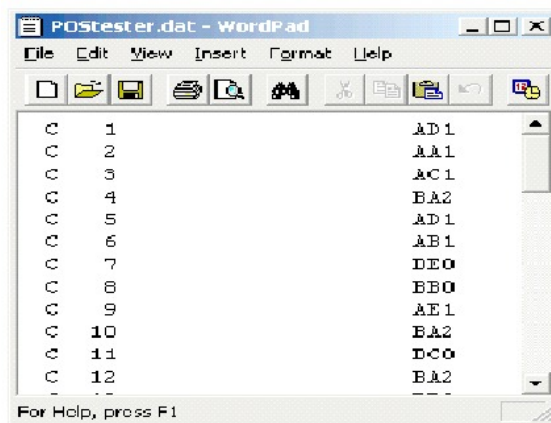
**Remarque :** La recherche n'est pas sensible à la casse.

## Exemples

Dans l'exemple ci-dessous, nous recherchons le modèle ou la chaîne 'A' dans la valeur de la colonne dbo\_BaseInfo.BranchCd et affectons la valeur renvoyée au fichier dérivé POSTest.



L'exemple suivant illustre quelques lignes de la table contenant les valeurs de dbo\_BaseInfo.BranchCd et POSTest côte-à-côte.



C	1	A.D 1
C	2	A.A 1
C	3	A.C 1
C	4	B.A.2
C	5	A.D 1
C	6	A.B 1
C	7	D.E 0
C	8	B.B 0
C	9	A.E 1
C	10	B.A.2
C	11	D.C 0
C	12	B.A.2

Exemple plus complexe :

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

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

Cette commande renvoie les lignes pour lesquelles les valeurs de CellCode se terminent par "AAA" après la deuxième occurrence de "X".

---

## Macro PLUS

est disponible dans IBM Campaign et IBM Interact. PLUS

### Syntaxe

data PLUS addend data + addend

### Paramètres

data

Plage de cibles contenant les nombres à ajouter. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

addend

Nombres à ajouter à toutes les valeurs de la colonne spécifiée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans data2 doit correspondre au nombre de colonnes dans data1, à moins que data2 ne soit une constante. Pour la définition du format de addend (identique à data), voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

PLUS ajoute les valeurs des deux plages de données spécifiées. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 ajoutée à la colonne correspondante de data2 (à savoir, la première colonne de data1 est ajoutée à la première colonne de data, la deuxième colonne à la deuxième colonne et ainsi de suite).

Si data2 est une constante, chaque valeur de data1 est augmentée de cette valeur. Si data2 contient une ou plusieurs colonnes, les calculs sont effectués ligne par ligne entre une colonne de data2 et une colonne de data2. La première ligne de data1 est ajoutée à la première ligne de data2, la seconde ligne, à la seconde ligne, etc. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** L'opérateur PLUS peut être abrégé en signe plus (+).

### Exemples

TEMP = 3 PLUS 4 ou TEMP = 3 + 4
---------------------------------

Crée une colonne nommée TEMP contenant la valeur sept.
--

TEMP = V1 + 8
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au contenu de la colonne V1 plus huit.
TEMP = V1 + V1
Crée une colonne nommée TEMP contenant deux fois le contenu de la colonne V1.
TEMP = V1 + V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 plus la valeur de ligne correspondante de la colonne V2.
TEMP = V1:V3 + V4:V6
Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 plus les valeurs de ligne correspondantes de la colonne V4. La colonne VX ajoute les colonnes V2 et V5. La colonne VY calcule la somme des colonnes V5 et V6.
TEMP = V1[10:20] + V2 ou TEMP = V1[10:20] + V2[1:11]
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les sommes des valeurs des lignes 10 à 20 de la colonne v1 et les valeurs des lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
MINUS	Soustrait une plage de données spécifiée d'une autre
SUM ou TOTAL	Calcule la somme d'une plage de cibles

---

## Macro POW

est disponible dans IBM Campaign et IBM Interact. POW

### Syntaxe

base POW exponent base ^ exponent

### Paramètres

base

Valeurs numériques à élever à une puissance exponentielle. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de base (identique à data), voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

exponent

Nombres exponentiels desquels les valeurs de data doivent être élevées. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans exponent doit correspondre au nombre de colonnes dans base, à moins que base ne soit une constante. Pour la définition du format de exponent (identique à data),

voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

POW élève les valeurs de la première plage de données à la puissance spécifiée dans la deuxième plage de données (il calcule  $base^{exponent}$ ). Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant le résultat de l'élévation de base à la puissance exponent (à savoir, la première colonne de data1 est élevée à la première colonne de data, la deuxième colonne à la deuxième colonne et ainsi de suite).

Si exponent est une constante, chaque valeur de base est élevée par cette valeur. Si exponent contient une ou plusieurs colonnes, les calculs sont effectués ligne par ligne entre une colonne de base et une colonne de exponent. La première ligne de base est élevée à la première ligne de exponent, la seconde ligne, à la seconde ligne, etc. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** L'opérateur POW peut être abrégé en accent circonflexe (^). Par exemple,  $TEMP = 2^8$  est équivalent à  $TEMP = 2 \text{ POW } 8$ .

**Remarque :** Si la valeur  $x$  est trop faible, une erreur de dépassement est renvoyée. Cela se produit si  $base^{exponent}$  dépasse la valeur à virgule flottante sur 32 bits maximale ou minimale.

## Exemples

$TEMP = 2 \text{ POW } 3$  ou  $TEMP = 2^3$

Crée une colonne nommée TEMP contenant la valeur huit.

$TEMP = V1 \wedge 0.5$

Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond à la racine carrée du contenu de la colonne V1 (équivalent à  $\text{SQRT}(V1)$ ).

$TEMP = V1 \wedge V3$

Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 élevée à la valeur de ligne correspondante de la colonne V2.

$TEMP = V1:V3 \wedge V4:V6$

Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 élevées aux valeurs de ligne correspondantes de la colonne V4. La colonne VX contient le résultat de la colonne V2 élevée aux valeurs correspondantes dans la colonne V5. La colonne VY contient le résultat de la colonne V3 élevée aux valeurs correspondantes de V6.

$TEMP = V1[10:20] \text{ POW } V2$  ou  $TEMP = V1[10:20] \text{ POW } V2[1:11]$

Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent le résultat de l'élévation des valeurs des lignes 10 à 20 de la colonne V1 par les valeurs des lignes 1 à 10 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
EXP	Calcule le nombre naturel (e) élevé à la puissance du contenu de chaque cible de la plage de données spécifiée
LN ou LOG	Calcule le logarithme naturel du contenu de la plage de données spécifiée
LN2	Calcule le logarithme base 2 du contenu de la plage de données spécifiée
LN10	Calcule le logarithme base 10 du contenu de la plage de données spécifiée

---

## Macro RANDOM

est disponible uniquement dans IBM Campaign. RANDOM

### Syntaxe

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

### Paramètres

num

Nombre de nombre aléatoires à générer. Il doit s'agir d'un entier positif supérieur à zéro.

value1

Limite sur les nombres aléatoires à générer. Il peut s'agir de toute valeur constante ou d'une expression convertie en constante. Si ce paramètre n'est pas fourni, la valeur par défaut est zéro.

value2

L'autre limite sur les nombres aléatoires à générer. Il peut s'agir de toute valeur constante ou d'une expression convertie en constante. Si ce paramètre n'est pas fourni, la valeur par défaut est un.

seed

Valeur de départ facultative à utiliser pour la génération de nombres aléatoires. Il doit s'agir d'un entier.

### Description

RANDOM génère une colonne de nombres aléatoires. Il renvoie une nouvelle colonne contenant num nombre aléatoires. Si value1 et value2 sont spécifiés, les nombres aléatoires sont générés entre ces limites (incluses). S'ils ne sont pas spécifiés, les valeurs sont générées par défaut entre zéro et un. Si seed est fourni, il est utilisé comme valeur de départ pour le générateur de nombres aléatoires.

**Remarque :** Si seed est supérieur ou égal à  $2^{32}$ , la valeur est remplacée par  $2^{32} - 1$ . Les valeurs de seed supérieures à  $2^{24}$  sont arrondies (la précision est perdue). Par conséquent, plusieurs valeurs peuvent posséder la même valeur seed.

## Exemples

TEMP = RANDOM()	Crée une colonne nommée TEMP contenant des nombres aléatoires de longueur illimitée.
TEMP = RANDOM(100)	Crée une colonne nommée TEMP contenant 100 nombres aléatoires compris entre 0,0 et 1,0.
TEMP = RANDOM(100, 5943049)	Crée une colonne nommée TEMP contenant 100 nombres aléatoires générés à partir du numéro de valeur de départ 5943049.
TEMP = RANDOM(100, 0, 100)	Crée une colonne nommée TEMP contenant 100 nombres aléatoires compris entre 0 et 100,0.
TEMP = RANDOM(100, 0, 100, 5943049)	Crée une colonne nommée TEMP contenant 100 nombres aléatoires compris entre 0 et 100 générés à partir du numéro de valeur de départ 5943049.

## Fonctions associées

Fonction	Description
RANDOM_GAUSS	Renvoie le nombre spécifié de valeurs aléatoires à partir d'une distribution de Gauss

---

## Macro RANDOM\_GAUSS

est disponible uniquement dans IBM Campaign. RANDOM\_GAUSS

### Syntaxe

RANDOM\_GAUSS(num [, seed]) RANDOM\_GAUSS(num, mean, std [, seed])

### Paramètres

num

Nombre de nombre aléatoires à générer. Il doit s'agir d'un entier positif supérieur à zéro.

mean

Moyenne de la valeur de Gauss. Il peut s'agir de toute valeur constante ou d'une expression convertie en constante. Si ce paramètre n'est pas fourni, la valeur par défaut est zéro.

std

Ecart type de la valeur de Gauss. Il peut s'agir de toute valeur constante ou d'une expression convertie en constante. Si ce paramètre n'est pas fourni, la valeur par défaut est un.

seed

Valeur de départ facultative à utiliser pour la génération de nombres aléatoires. Il doit s'agir d'un entier. (Si une valeur non entière est fournie, la partir entière de cette valeur est automatiquement utilisée à la place.)

## Description

RANDOM\_GAUSS génère une colonne de nombres aléatoires basée sur une distribution de Gauss. Il renvoie une nouvelle colonne contenant num nombre aléatoires. Si mean et std sont spécifiés, les nombres aléatoires sont générés à l'aide d'une distribution de Gauss avec la moyenne et l'écart type spécifiés. S'ils ne sont pas spécifiés, la valeur de Gauss par défaut possède une moyenne de zéro et un écart type d'un. Si seed est fourni, il est utilisé comme valeur de départ pour le générateur de nombres aléatoires.

## Exemples

TEMP = RANDOM_GAUSS(100)
Crée une colonne nommée TEMP contenant 100 valeurs échantillonnées de manière aléatoire à partir d'une distribution de Gauss de moyenne nulle et d'écart type égal à un.
TEMP = RANDOM_GAUSS(500, 3)
Crée une colonne nommée TEMP contenant 100 valeurs échantillonnées de manière aléatoire à partir d'une distribution de Gauss de moyenne nulle et d'écart type égal à un. Le nombre 3 est utilisé comme valeur de départ pour le générateur de nombre aléatoire.
TEMP = RANDOM_GAUSS(5000, 100, 32)
Crée une colonne nommée TEMP contenant 5000 valeurs échantillonnées de manière aléatoire à partir d'une distribution de Gauss de moyenne égale à 100 et d'écart type égal à 32.
TEMP = RANDOM_GAUSS(500, -1, 2, 3)
Crée une colonne nommée TEMP contenant 500 valeurs échantillonnées de manière aléatoire à partir d'une distribution de Gauss de moyenne égale à -1 et d'écart type égal à 2. Le nombre 3 est utilisé comme valeur de départ pour le générateur de nombre aléatoire.

## Fonctions associées

Fonction	Description
RANDOM	Renvoie le nombre spécifié de nombres aléatoires

---

## Macro ROUND

est disponible uniquement dans IBM Campaign. ROUND

### Syntaxe

ROUND(data)



## Paramètres

data

Valeurs numériques à arrondir. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

ROUND arrondit les valeurs de la plage de données spécifiée à l'entier le plus proche. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant la valeur arrondie des nombres de la colonne d'entrée correspondante. Les nombres à égale distance de deux entiers sont arrondis vers le haut (par exemple, 2.5 est arrondi à 3.0 et -2.5 est arrondi à -2.0).

## Exemples

TEMP = ROUND(3.2)
Crée une colonne nommée TEMP contenant la valeur trois.
TEMP = ROUND(V1)
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond à la valeur arrondie du contenu de la colonne V1.
TEMP = ROUND(V1:V3)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux valeurs arrondies du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux valeurs arrondies du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux valeurs arrondies du contenu de la colonne V3.
TEMP = ROUND(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les valeurs arrondies des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.
TEMP = ROUND(V1[1:5]:V2)
Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux valeurs arrondies des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux valeurs arrondies des lignes correspondantes de la colonne V2.

## Fonctions associées

Fonction	Description
INT	Calcule la valeur entière (arrondie) du contenu de la plage de données spécifiée
MOD	Calcule le modulo du contenu de la plage de données spécifiée
TRUNCATE	Renvoie la partie non décimale de chaque valeur de la plage de données spécifiée

---

## Macro ROWNUM

est disponible uniquement dans IBM Campaign. ROWNUM

### Syntaxe

ROWNUM()

### Description

ROWNUM génère des nombres séquentiels de un à n, où n correspond au nombre d'enregistrements. Le nombre pour le premier enregistrement est 1, celui du deuxième 2 et ainsi de suite

**Remarque :** Le nombre maximal d'enregistrements que ROWNUM peut gérer est de deux milliards.

---

## Macro RTRIM

La macro RTRIM est disponible dans IBM Campaign et IBM Interact.

### Syntaxe

RTRIM(data)

### Paramètres

data

### Description

RTRIM supprime les derniers caractères d'espace de chaque valeur de chaîne contenue dans la plage de données spécifiée, puis renvoie la chaîne convertie. Il renvoie une nouvelle colonne pour chaque colonne d'entrée.

### Exemples

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

Crée une chaîne nommée Temp qui contient "gold".

---

## Macro SIGN

est disponible uniquement dans IBM Campaign. SIGN

### Syntaxe

SIGN(data)

Paramètres

data

Valeurs numériques dont la signe doit être déterminé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie

comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

SIGN teste le signe des valeurs de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant le signe des nombres de la colonne d'entrée correspondante. Un un positif est renvoyé pour toutes les valeurs supérieures à zéro ; un un négatif est renvoyé pour toutes les valeurs inférieures à zéro ; zéro est renvoyé pour les valeurs égales à zéro.

## Exemples

TEMP = SIGN(-3)
Crée une colonne nommée TEMP contenant la valeur -1.
TEMP = SIGN(V1)
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au signe du contenu de la colonne V1.
TEMP = SIGN(V1:V3)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux signes du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux signes du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux signes du contenu de la colonne V3.
TEMP = SIGN(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les signes des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.
TEMP = SIGN(V1[10:50]:V2)
Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 41 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux sinus des valeurs des lignes 10 à 50 de la colonne V1 et les valeurs de la colonne VX correspondent aux signes des valeurs des lignes 10 à 50 de la colonne V2.

## Fonctions associées

Fonction	Description
ABS	Calcule la valeur absolue du contenu de la plage de données spécifiée
INVERSE	Calcule la valeur négative du contenu de la plage de données spécifiée

---

## Macro SIN

est disponible uniquement dans IBM Campaign. SIN

### Syntaxe

SIN(data [, units\_keyword])

## Paramètres

data

Valeurs numériques dont le sinus doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

units\_keyword

Ce mot clé facultatif détermine si les valeurs en entrée et les résultats sont interprétés comme des degrés ou des radians. Sélectionnez l'une des valeurs suivantes :

RADIAN - Effectue les calculs en radians (valeur par défaut)

DEGREE - Effectue les calculs en degrés

Si ce paramètre n'est pas spécifié, la valeur par défaut est radians. (Pour convertir des radians en degrés, divisez par PI et multipliez par 180.)

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

## Description

SIN calcule le sinus des valeurs de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant le sinus des nombres de la colonne d'entrée correspondante.

## Exemples

```
TEMP = SIN(PI/2), TEMP = SIN(PI/2, 0) ou TEMP = SIGN(PI/2, RADIAN)
```

Crée une colonne nommée TEMP contenant la valeur un.

```
TEMP = SIN(V1)
```

Crée une colonne nommée TEMP, où chaque valeur correspond au sinus (en radians) du contenu de la colonne V1.

```
TEMP = SIN(V1:V3, 1) ou TEMP = SIN(V1:V3, DEGREE)
```

Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux sinus du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux sinus du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux sinus du contenu de la colonne V3. Toutes les valeurs sont en degrés.

```
TEMP = SIN(V1[10:50]:V2)
```

Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 41 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux sinus des valeurs des lignes 10 à 50 de la colonne V1 et les valeurs de la colonne VX correspondent aux sinus des valeurs des lignes 10 à 50 de la colonne V2. Toutes les valeurs sont en radians.

## Fonctions associées

Fonction	Description
ASIN	Calcule l'arc sinus du contenu de la plage de données spécifiée
COS	Calcule le cosinus du contenu de la plage de données spécifiée
SINH	Calcule le sinus hyperbolique du contenu de la plage de données spécifiée
TAN	Calcule la tangente du contenu de la plage de données spécifiée

---

## Macro SINH

est disponible uniquement dans IBM Campaign. SINH

### Syntaxe

SINH(data [, units\_keyword])

### Paramètres

data

Valeurs numériques dont le sinus hyperbolique doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

units\_keyword

Ce mot clé facultatif détermine si les valeurs en entrée et les résultats sont interprétés comme des degrés ou des radians. Sélectionnez l'une des valeurs suivantes :

RADIAN - Effectue les calculs en radians (valeur par défaut)

DEGREE - Effectue les calculs en degrés

Si ce paramètre n'est pas spécifié, la valeur par défaut est radians. (Pour convertir des radians en degrés, divisez par PI et multipliez par 180.)

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

### Description

SINH calcule le sinus hyperbolique des valeurs de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant le sinus hyperbolique des nombres de la colonne d'entrée correspondante. Pour  $x$  en radians, le sinus hyperbolique d'un nombre est :

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

où  $e$  représente le nombre naturel, 2.7182818.

**Remarque :** Si la valeur  $x$  est trop élevée, une erreur de dépassement est renvoyée. Cela se produit si  $\sinh(x)$  dépasse la valeur maximale à virgule flottante sur 32 bits.

## Exemples

TEMP = SINH(1), TEMP = SINH(1, 0) ou TEMP = SINH(1, RADIAN)
Crée une colonne nommée TEMP contenant la valeur 1.18.
TEMP = SINH(V1)
Crée une colonne nommée TEMP, où chaque valeur correspond au sinus hyperbolique (en radians) du contenu de la colonne V1.
TEMP = SINH(V1:V3, 1) ou TEMP = SINH(V1:V3, DEGREE)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux sinus hyperboliques du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux sinus hyperboliques du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux sinus hyperboliques du contenu de la colonne V3. Toutes les valeurs sont en degrés.
TEMP = SINH(V1[10:50]:V2)
Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 41 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux sinus hyperboliques des valeurs des lignes 10 à 50 de la colonne V1 et les valeurs de la colonne VX correspondent aux sinus hyperboliques des valeurs des lignes 10 à 50 de la colonne V2. Toutes les valeurs sont en radians.

## Fonctions associées

Fonction	Description
COSH	Calcule le cosinus hyperbolique du contenu de la plage de données spécifiée
SIN	Calcule le sinus du contenu de la plage de données spécifiée
TANH	Calcule la tangente hyperbolique du contenu de la plage de données spécifiée

---

## Macro SQRT

est disponible uniquement dans IBM Campaign. SQRT

## Syntaxe

SQRT(data)

Paramètres

data

Valeurs numériques dont la racine carrée doit être calculée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

SQRT calcule la racine carrée des valeurs de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant la racine carrée positive des nombres de la colonne d'entrée correspondante.

**Remarque :** Si l'une des valeurs de la plage de données définie est négative, le symbole ??? est renvoyé pour cette cible.

## Exemples

TEMP = SQRT(2)
Crée une colonne nommée TEMP contenant la valeur 1.41.
TEMP = SQRT(V1)
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond à la racine carrée du contenu de la colonne V1.
TEMP = SQRT(V1:V3)
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux racines carrées du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux racines carrées du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux racines carrées du contenu de la colonne V3.
TEMP = SQRT(V1[10:20])
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les racines carrées des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.
TEMP = SQRT(V1[10:50]:V2)
Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 41 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux racines carrées des valeurs des lignes 10 à 50 de la colonne V1 et les valeurs de la colonne VX correspondent aux racines carrées des valeurs des lignes 10 à 50 de la colonne V2.

## Fonctions associées

Fonction	Description
DIV	Divise une plage de données spécifiée par une autre

MULT	Multiplie le contenu de deux plages de données
POW	Calcule une valeur de base élevée aux puissances exponentielles spécifiées

---

## Macro STDV ou STDEV

est disponible dans IBM Campaign et IBM Interact. STDV ou STDEV

### Syntaxe

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

### Paramètres

data

Valeurs numériques dont l'écart type doit être calculé. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

keyword

Ce mot clé facultatif détermine la manière dont le calcul est effectué sur la plage de données d'entrée. Sélectionnez l'une des valeurs suivantes :

ALL - Effectue le calcul sur toutes les cibles de data (valeur par défaut)

COL - Effectue le calcul séparément pour chaque colonne de data

ROW - Effectue le calcul séparément pour chaque ligne de data

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

**Remarque :** De nombreuses fonctions de macro utilisent les paramètres de mot clé {ALL | COL | ROW}. Ces mots clés ne s'appliquent pas dans **IBM Campaign** car les données d'entrées correspondent toujours à une colonne ou à une zone. La macro se comporte toujours comme si le mot clé COL a été spécifié. Par conséquent, vous n'avez pas besoin de spécifier ces mots clés lorsque vous utilisez **IBM Campaign**.

### Description

STDV calcule l'écart type de toutes les cibles de la plage de données spécifiée. L'écart type d'une distribution est égal à la racine carrée de la variance. L'écart type est calculé comme suit :



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

où les  $x$  représentent les échantillons,  $n$  représente le nombre d'échantillons et  $\text{mean}$  est la moyenne de la distribution.

**Remarque :** Si le nombre d'échantillons  $n = 1$ , STDV renvoie une erreur.

## Exemples

TEMP = STDV(V1)
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à l'écart type du contenu de la colonne V1.
TEMP = STDV(V1:V3)
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à l'écart type du contenu des colonnes V1, V2 et V3.
TEMP = STDV(V1[1:5]:V4)
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à l'écart type des cibles des lignes 1 à 5 des colonnes V1 à V4.
TEMP = STDV(V1:V3, COL)
Crée trois colonnes nommées TEMP, VX et VY. La valeur unique de la colonne TEMP correspond à l'écart type du contenu de la colonne V1, la valeur unique de la colonne VX correspond à l'écart type du contenu de la colonne V2 et la valeur unique de la colonne VY correspond à l'écart type du contenu de la colonne V3.
TEMP = STDV(V1[10:50]:V3,COL)
Crée trois colonnes nommées TEMP, VX et VY contenant chacune une seule valeur. La valeur de la colonne TEMP correspond à l'écart type des cibles des lignes 10 à 50 de la colonne V1, la valeur de la colonne VX correspond à l'écart type des cibles des lignes 10 à 50 de la colonne V2 et la valeur de la colonne VY correspond à l'écart type des cibles des lignes 10 à 50 de la colonne V3.
TEMP = STDV(V1:V3, ROW)
Crée une colonne nommée TEMP dans laquelle chaque entrée de cible correspond à l'écart type de la ligne correspondante dans les colonnes V1, V2 et V3.
TEMP = STDV(V1[1:5]:V3,ROW)
Crée une colonne nommée TEMP, dans laquelle les cibles des lignes 1 à 5 contiennent l'écart type de la ligne correspondante dans les colonnes V1 à V3. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
VAR	Calcule la variance d'une plage de cibles

---

## Macro **STRING\_CONCAT**

est disponible dans IBM Campaign et IBM Interact. **STRING\_CONCAT**

### Syntaxe

```
STRING_CONCAT(string1, string2, ... stringN)
```

### Paramètres

string

Chaîne de texte ASCII à concaténer. Il peut s'agir de texte ASCII entre guillemets, d'une colonne de texte, d'une plage de cibles contenant du texte ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de string (identique à data), voir la section *Paramètres des fonctions macro* du chapitre *Utilisation de macros* de votre produit.

### Description

**STRING\_CONCAT** concatène les valeurs de texte ASCII de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant les chaînes concaténées des lignes correspondantes de strings. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** La longueur totale de chaque chaîne résultante ne peut pas dépasser 255 caractères.

IBM Interact prend également en charge la syntaxe suivante :

```
STRING_CONCAT( string1 , string2 , ... stringN )
```

Par exemple, **STRING\_CONCAT('a', 'b', 'c', 'd')** est valide.

### Exemples

<pre>TEMP = STRING_CONCAT("house", "boat")</pre>
Crée une colonne nommée TEMP, qui contient la chaîne de texte ASCII " houseboat ".
<pre>TEMP = STRING_CONCAT(V1, ".")</pre>
Crée une colonne nommée TEMP, chacune de ses lignes contenant la chaîne de texte ASCII de la ligne correspondante de la colonne V1 à laquelle un point a été ajouté.
<pre>TEMP = STRING_CONCAT(V1, V2)</pre>
Crée une colonne nommée TEMP, chacune de ses lignes contenant la chaîne de texte ASCII de la colonne V1 concaténée avec la chaîne de texte de la colonne V2.
<pre>TEMP = STRING_CONCAT(V1:V3, V4:V6)</pre>
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux chaînes concaténées des lignes correspondantes des colonnes V1 et V4, les valeurs de la colonne VX correspondent aux chaînes concaténées des lignes correspondantes des colonnes V2 et V5 et les valeurs de la colonne VY correspondent aux chaînes concaténées des lignes correspondantes des colonnes V3 et V6.

<pre>TEMP = STRING_CONCAT(V1[5:10]:V2, V3:V4)</pre> <p>Crée deux colonnes nommées TEMP et VX. Les valeurs de la colonne TEMP correspondent aux chaînes des lignes 5 à 10 de la colonne V1 concaténées avec les lignes 1 à 6 de la colonne V3. Les valeurs de VX correspondent aux chaînes des lignes 5 à 10 de la colonne V2 concaténées avec les lignes 1 à 6 de la colonne V4.</p>
<pre>TEMP = STRING_CONCAT('a', 'b', 'c', 'd')</pre> <p>Crée une colonne nommée TEMP, qui contient la chaîne de texte ASCII "abcd".</p>

## Fonctions associées

Fonction	Description
STRING_HEAD	Renvoie les n premiers caractères de chaque chaîne de la plage de données spécifiée
STRING_LENGTH	Renvoie la longueur de chaque chaîne de la plage de données spécifiée
STRING_SEG	Renvoie le segment de chaîne entre deux index spécifiés
STRING_TAIL	Renvoie les n derniers caractères de chaque chaîne de la plage de données spécifiée

---

## Macro STRING\_HEAD

est disponible uniquement dans IBM Campaign. STRING\_HEAD

### Syntaxe

```
STRING_HEAD(num_chars, data)
```

### Paramètres

num\_chars

Nombre de caractères à renvoyer à partir du début de chaque chaîne de data. Il doit s'agir d'un entier positif supérieur à zéro.

data

Valeurs de chaîne de texte ASCII. Il peut s'agir de texte ASCII entre guillemets, d'une colonne de texte, d'une plage de cibles contenant du texte ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

STRING\_HEAD renvoie les num\_chars premiers caractères de chaque valeur de chaîne de la plage de données spécifiée. Si num\_chars est supérieur au nombre de caractères d'une chaîne de texte, les caractères restants sont remplacés par le caractère null " \0 ".

## Exemples

<pre>TEMP = STRING_HEAD(3, "JAN 15, 1997")</pre>
Crée une colonne nommée TEMP, qui contient la chaîne de texte ASCII " JAN ".
<pre>TEMP = STRING_HEAD(10, "Pressure")</pre>
Crée une colonne nommée TEMP, qui contient la chaîne de texte ASCII " Pressure ".
<pre>TEMP = STRING_HEAD(5, V1)</pre>
Crée une colonne nommée TEMP contenant les cinq premiers caractères de chaque chaîne de la colonne V1.
<pre>TEMP = STRING_HEAD(1, V1:V3)</pre>
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux premiers caractères des chaînes des lignes correspondantes de la colonne V1, les valeurs de la colonne VX correspondent aux premiers caractères des chaînes des lignes correspondantes de la colonne V2 et les valeurs de la colonne VY correspondent aux premiers caractères des chaînes des lignes correspondantes de la colonne V3.
<pre>TEMP = STRING_HEAD(12, V4[1:50]:V6]</pre>
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux 12 premiers caractères des chaînes des lignes 1 à 50 de la colonne V1, les valeurs de la colonne VX correspondent aux 12 premiers caractères des chaînes des lignes 1 à 50 de la colonne V2 et les valeurs de la colonne VY correspondent aux 12 premiers caractères des chaînes des lignes 1 à 50 de la colonne V3.

## Fonctions associées

Fonction	Description
STRING_CONCAT	Concatène deux chaînes de texte des plages de données spécifiées
STRING_LENGTH	Renvoie la longueur de chaque chaîne de la plage de données spécifiée
STRING_SEG	Renvoie le segment de chaîne entre deux index spécifiés
STRING_TAIL	Renvoie les n derniers caractères de chaque chaîne de la plage de données spécifiée

---

## Macro STRING\_LENGTH

est disponible uniquement dans IBM Campaign. STRING\_LENGTH

### Syntaxe

```
STRING_LENGTH(data)
```

Paramètres

data

Valeurs de chaîne de texte ASCII dont la longueur doit être calculée. Il peut s'agir de texte ASCII entre guillemets, d'une colonne de texte, d'une plage de cibles contenant du texte ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro"

du chapitre du présent document correspondant à votre produit IBM.

## Description

STRING\_LENGTH renvoie la longueur de chaque valeur de chaîne dans la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant la longueur de la chaîne de texte correspondante.

**Remarque :** Si STRING\_LENGTH est appliqué à des colonnes contenant des données numériques, il renvoie des zéros.

## Exemples

TEMP = STRING_LENGTH("four") Crée une colonne nommée TEMP contenant la valeur 4.
TEMP = STRING_LENGTH(4) Crée une colonne nommée TEMP contenant la valeur 0.
TEMP = STRING_LENGTH(V1) Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond à la longueur de la chaîne dans la ligne correspondante de la colonne V1.
TEMP = STRING_LENGTH(V1:V3) Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux longueurs des chaînes des lignes correspondantes de la colonne V1, les valeurs de la colonne VX correspondent aux longueurs des chaînes des lignes correspondantes de la colonne V2 et les valeurs de la colonne VY correspondent aux longueurs des chaînes des lignes correspondantes de la colonne V3.
TEMP = STRING_LENGTH(V4[1:50]:V6) Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux longueurs des chaînes des lignes 1 à 50 de la colonne V1, les valeurs de la colonne VX correspondent aux longueurs des chaînes des lignes 1 à 50 de la colonne V2 et les valeurs de la colonne VY correspondent aux longueurs des chaînes des lignes 1 à 50 de la colonne V3.

## Fonctions associées

Fonction	Description
STRING_CONCAT	Concatène deux chaînes de texte des plages de données spécifiées
STRING_HEAD	Renvoie les n premiers caractères de chaque chaîne de la plage de données spécifiée
STRING_SEG	Renvoie le segment de chaîne entre deux index spécifiés
STRING_TAIL	Renvoie les n derniers caractères de chaque chaîne de la plage de données spécifiée

---

## Macro STRING\_PROPER

est disponible uniquement dans IBM Campaign. STRING\_PROPER

## Syntaxe

STRING\_PROPER(data)

## Paramètres

data

Valeur de chaîne à convertir.

## Description

STRING\_PROPER convertit chaque valeur de chaîne contenue dans la plage de données spécifiée en mettant en majuscule la première lettre de chaque chaîne ou toute lettre suivant un espace blanc ou un symbole (autre que le trait de soulignement) et transforme en minuscules tous les autres caractères. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant la chaîne convertie dans la colonne d'entrée correspondante.

## Exemples

Temp = STRING_PROPER
----------------------

---

## Macro STRING\_SEG

est disponible uniquement dans IBM Campaign. STRING\_SEG

## Syntaxe

STRING\_SEG(from, to, data)

## Paramètres

from

Nombre de caractères depuis le début de la chaîne à partir duquel le segment de chaîne doit commencer à être extrait. Il doit s'agir d'un entier positif supérieur à zéro et inférieur à to ou STRING\_SEG renvoie une chaîne vide.

to

Nombre de caractères depuis le début de la chaîne à partir duquel le segment de chaîne ne doit plus être extrait. Il doit s'agir d'un entier positif supérieur ou égal à from. Si to est égal à from (et to est inférieur ou égal à la longueur de la chaîne), un caractère est renvoyé.

data

Valeurs de chaîne de texte ASCII. Il peut s'agir de texte ASCII entre guillemets, d'une colonne de texte, d'une plage de cibles contenant du texte ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

STRING\_SEG renvoie le segment de chaîne entre deux index de chaque valeur de chaîne dans la plage de données spécifiée. Si from est supérieur à la longueur d'une chaîne, aucune valeur n'est renvoyée. Si to est supérieur à la longueur d'une chaîne, tous les caractères de from sont renvoyés.

## Exemples

<pre>TEMP = STRING_SEG(1, 6, "JAN 15, 1997")</pre>
Crée une colonne nommée TEMP, qui contient la chaîne de texte ASCII " Jan 15 ".
<pre>TEMP = STRING_SEG(5, 20, "Pressure")</pre>
Crée une colonne nommée TEMP, qui contient la chaîne de texte ASCII " sure ".
<pre>TEMP = STRING_SEG(5, 6, V1)</pre>
Crée une colonne nommée TEMP contenant les cinquième et sixième caractères de chaque chaîne de la colonne V1.
<pre>TEMP = STRING_SEG(10, 20, V1:V3)</pre>
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux caractères 10 à 20 des chaînes des lignes correspondantes de la colonne V1, les valeurs de la colonne VX correspondent aux caractères 10 à 20 des chaînes des lignes correspondantes de la colonne V2 et les valeurs de la colonne VY correspondent aux caractères 10 à 20 des chaînes des lignes correspondantes de la colonne V3.
<pre>TEMP = STRING_SEG(5, 10, V4[1:50]:V6]</pre>
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux caractères 5 à 10 des chaînes des lignes 1 à 50 de la colonne V1, les valeurs de la colonne VX correspondent aux caractères 5 à 10 des chaînes des lignes 1 à 50 de la colonne V2 et les valeurs de la colonne VY correspondent aux caractères 5 à 10 des chaînes des lignes 1 à 50 de la colonne V3.

## Fonctions associées

Fonction	Description
STRING_CONCAT	Concatène deux chaînes de texte des plages de données spécifiées
STRING_HEAD	Renvoie les n premiers caractères de chaque chaîne de la plage de données spécifiée
STRING_LENGTH	Renvoie la longueur de chaque chaîne de la plage de données spécifiée
STRING_TAIL	Renvoie les n derniers caractères de chaque chaîne de la plage de données spécifiée

---

## Macro STRING\_TAIL

est disponible uniquement dans IBM Campaign. STRING\_TAIL

### Syntaxe

STRING\_TAIL(num\_chars, data)

## Paramètres

num\_chars

Nombre de caractères à renvoyer à partir de la fin de chaque chaîne de data. Il doit s'agir d'un entier positif supérieur à zéro.

data

Valeurs de chaîne de texte ASCII. Il peut s'agir de texte ASCII entre guillemets, d'une colonne de texte, d'une plage de cibles contenant du texte ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

## Description

STRING\_TAIL renvoie les num\_chars derniers caractères de chaque valeur de chaîne de la plage de données spécifiée. Toutes les valeurs de chaîne sont remplies jusqu'à la longueur de la plus longue chaîne avec des caractères nuls "\0". Ensuite, les num\_chars derniers caractères sont renvoyés de chaque chaîne. Si num\_chars est supérieur au nombre de caractères d'une chaîne de texte, l'intégralité de la chaîne de texte est renvoyée.

## Exemples

<pre>TEMP = STRING_TAIL(3, "JAN 15, 1997")</pre>
Crée une colonne nommée TEMP, qui contient la chaîne de texte ASCII " 997 ".
<pre>TEMP = STRING_TAIL(10, "Pressure")</pre>
Crée une colonne nommée TEMP, qui contient la chaîne de texte ASCII " Pressure ".
<pre>TEMP = STRING_TAIL(5, V1)</pre>
Crée une colonne nommée TEMP contenant les cinq derniers caractères de chaque chaîne de la colonne V1.
<pre>TEMP = STRING_TAIL(1, V1:V3)</pre>
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux derniers caractères des chaînes des lignes correspondantes de la colonne V1, les valeurs de la colonne VX correspondent aux derniers caractères des chaînes des lignes correspondantes de la colonne V2 et les valeurs de la colonne VY correspondent aux derniers caractères des chaînes des lignes correspondantes de la colonne V3.
<pre>TEMP = STRING_TAIL(12, V4[1:50]:V6]</pre>
Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux 12 derniers caractères des chaînes des lignes 1 à 50 de la colonne V1, les valeurs de la colonne VX correspondent aux 12 derniers caractères des chaînes des lignes 1 à 50 de la colonne V2 et les valeurs de la colonne VY correspondent aux 12 derniers caractères des chaînes des lignes 1 à 50 de la colonne V3.

## Fonctions associées

Fonction	Description
STRING_CONCAT	Concatène deux chaînes de texte des plages de données spécifiées



Fonction	Description
STRING_HEAD	Renvoie les n premiers caractères de chaque chaîne de la plage de données spécifiée
STRING_LENGTH	Renvoie la longueur de chaque chaîne de la plage de données spécifiée
STRING_SEG	Renvoie le segment de chaîne entre deux index spécifiés

---

## Macro SUBSTR ou SUBSTRING

est disponible dans IBM Campaign et IBM Interact. SUBSTR ou SUBSTRING

### Syntaxe

SUBSTR(string\_value, start\_pos[, nchars]) or SUBSTR(string\_value FROM start\_pos[ FOR nchars]) SUBSTRING(string\_value, start\_pos[, nchars]) or SUBSTRING(string\_value FROM start\_pos[ FOR nchars])

### Paramètres

string\_value

Chaîne à partir de laquelle une sous-chaîne est extraite.

start\_pos

Caractère de début à partir duquel la sous-chaîne doit être extraite.

nchars

Nombre de caractères à extraire (doit être supérieur ou égal à 0). Si cette valeur n'est pas fournie, tous les caractères restants de string\_value sont extraits.

### Description

SUBSTR ou SUBSTRING extrait nchars caractères de la chaîne, à partir de start\_pos. Si nchars est omis, SUBSTR et SUBSTRING extraient les caractères de start\_pos jusqu'à la fin de la chaîne. Les espaces de fin sont automatiquement tronqués. Pour éviter les erreurs de syntaxe, veillez à séparer les valeurs numériques par une virgule et un espace, comme le montrent les exemples.

**Important :** IBM Interact ne prend en charge que les formats suivants : SUBSTR(string\_value, start\_pos[, nchars]) ou SUBSTRING(string\_value, start\_pos[, nchars])

### Exemples

SUBSTR SUBSTR Renvoie	("abcdef" FROM 1 FOR 2) ("abcdef", 1, 2) 'ab'
SUBSTR SUBSTR Renvoie	("abcdef" FROM -2 FOR 4) ("abcdef", -2, 4) 'a'
SUBSTR SUBSTR Renvoie	("abcdef" FROM 3) ("abcdef",3) 'cdef'

---

## Macro SUM

est disponible dans IBM Campaign et IBM Interact. SUM

### Syntaxe

SUM(data [, keyword])

### Paramètres

data

Valeurs numériques dont la somme doit être calculée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

keyword

Ce mot clé facultatif détermine la manière dont le calcul est effectué sur la plage de données d'entrée. Sélectionnez l'une des valeurs suivantes :

ALL - Effectue le calcul sur toutes les cibles de data (valeur par défaut)

COL - Effectue le calcul séparément pour chaque colonne de data

ROW - Effectue le calcul séparément pour chaque ligne de data

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

**Remarque :** De nombreuses fonctions de macro utilisent les paramètres de mot clé {ALL | COL | ROW}. Ces mots clés ne s'appliquent pas dans IBM Campaign car les données d'entrées correspondent toujours à une colonne ou à une zone. La macro se comporte toujours comme si le mot clé COL a été spécifié. Par conséquent, vous n'avez pas besoin de spécifier ces mots clés lorsque vous utilisez IBM Campaign.

### Description

SUM calcule la somme de toutes les cibles de la plage de données spécifiée. Il renvoie une colonne unique.

**Remarque :** SUM est identique à la fonction macro TOTAL.

### Exemples

TEMP = SUM(3)Crée une colonne nommée TEMP contenant la valeur trois.
TEMP = SUM((COLUMN(3, 5, 1))
Crée une colonne nommée TEMP contenant la valeur neuf.
TEMP = SUM(V1)
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la somme du contenu de la colonne V1.

TEMP = SUM(V1:V3)
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la somme du contenu des colonnes V1, V2 et V3.
TEMP = SUM(V1[1:5]:V4)
Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la somme des cibles des lignes 10 à 20 des colonnes V1 à V4.
TEMP = SUM(V1:V3, COL)
Crée trois colonnes nommées TEMP, VX et VY. La valeur unique de la colonne TEMP correspond à la somme du contenu de la colonne V1, la valeur unique de la colonne VX correspond à la somme du contenu de la colonne V2 et la valeur unique de la colonne VY correspond à la somme du contenu de la colonne V3.
TEMP = SUM(V1[1:5]:V3, COL)
Crée trois colonnes nommées TEMP, VX et VY contenant chacune une seule valeur. La valeur de la colonne TEMP correspond à la somme des cibles des lignes 1 à 5 de la colonne V1, la valeur de la colonne VX correspond à la somme des cibles des lignes 1 à 5 de la colonne V2 et la valeur de la colonne VY correspond à la somme des cibles des lignes 1 à 5 de la colonne V3.
TEMP = SUM(V1:V3, ROW)
Crée une colonne nommée TEMP dans laquelle chaque entrée de cible correspond à la somme de la ligne correspondante dans les colonnes V1, V2 et V3.
TEMP = SUM(V1[1:5]:V3, ROW)
Crée une colonne nommée TEMP, dans laquelle les cibles des lignes 1 à 5 contiennent la somme de la ligne correspondante dans les colonnes V1 à V3. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
AVG ou MEAN	Calcule la moyenne arithmétique d'une plage de cibles

## Macro TAN

est disponible uniquement dans IBM Campaign. TAN

### Syntaxe

TAN(data [, units\_keyword])

### Paramètres

data

Valeurs numériques dont la tangente doit être calculée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

units\_keyword

Ce mot clé facultatif détermine si les valeurs en entrée et les résultats sont interprétés comme des degrés ou des radians. Sélectionnez l'une des valeurs suivantes :

RADIAN - Effectue les calculs en radians (valeur par défaut)

DEGREE - Effectue les calculs en degrés

Si ce paramètre n'est pas spécifié, la valeur par défaut est radians. (Pour convertir des radians en degrés, divisez par PI et multipliez par 180.)

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

## Description

TAN calcule la tangente des valeurs de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant la tangente des nombres de la colonne d'entrée correspondante.

## Exemples

TEMP = TAN(PI/4), TEMP = TAN(PI/4, 0) ou TEMP = TAN(PI/4, RADIAN) Crée une colonne nommée TEMP contenant la valeur un.
TEMP = TAN(V1) Crée une colonne nommée TEMP, où chaque valeur correspond à la tangente (en radians) du contenu de la colonne V1.
TEMP = TAN(V1:V3, 1) ou TEMP = TAN(V1:V3, DEGREE) Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux tangentes du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux tangentes du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux tangentes du contenu de la colonne V3. Toutes les valeurs sont en degrés.
TEMP = TAN(V1[1:5]:V2) Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux tangentes des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux tangentes des lignes correspondantes de la colonne V2. Toutes les valeurs sont en radians.

## Fonctions associées

Fonction	Description
ATAN	Calcule l'arc tangente du contenu de la plage de données spécifiée
COS	Calcule le cosinus du contenu de la plage de données spécifiée
COT	Calcule la cotangente du contenu de la plage de données spécifiée
SIN	Calcule le sinus du contenu de la plage de données spécifiée

Fonction	Description
TANH	Calcule la tangente hyperbolique du contenu de la plage de données spécifiée

---

## Macro TANH

La macro TANH est disponible uniquement dans IBM Campaign.

### Syntaxe

TANH(data [, units\_keyword])

### Paramètres

data

Valeurs numériques dont la tangente hyperbolique doit être calculée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

units\_keyword

Ce mot clé facultatif détermine si les valeurs en entrée et les résultats sont interprétés comme des degrés ou des radians. Sélectionnez l'une des valeurs suivantes :

RADIAN - Effectue les calculs en radians (valeur par défaut)

DEGREE - Effectue les calculs en degrés

Si ce paramètre n'est pas spécifié, la valeur par défaut est radians. (Pour convertir des radians en degrés, divisez par PI et multipliez par 180.)

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

### Description

TANH calcule la tangente hyperbolique des valeurs de la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant la tangente hyperbolique des nombres de la colonne d'entrée correspondante. La tangente hyperbolique d'un nombre est calculée comme suit :

$$\tanh(x) = \frac{\sinh(x)}{\cosh(x)}$$

**Remarque :** Si la valeur  $x$  est trop élevée, une erreur de dépassement est renvoyée. Cela se produit si  $\tanh(x)$  dépasse la valeur maximale à virgule flottante sur 32 bits. Si  $\cosh(x)$  est égal à zéro, TANH renvoie la valeur maximale à virgule flottante sur 32 bits.

## Exemples

$TEMP = \text{TANH}(PI)$ , $TEMP = \text{TANH}(PI, 0)$ ou $TEMP = \text{TANH}(PI, \text{RADIAN})$ Crée une colonne nommée TEMP contenant la valeur un.
$TEMP = \text{TANH}(V1)$ Crée une colonne nommée TEMP, où chaque valeur correspond à la tangente hyperbolique (en radians) du contenu de la colonne V1.
$TEMP = \text{TANH}(V1:V3, 1)$ ou $TEMP = \text{TANH}(V1:V3, \text{DEGREE})$ Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux tangentes hyperboliques du contenu de la colonne V1, les valeurs de la colonne VX correspondent aux tangentes hyperboliques du contenu de la colonne V2 et les valeurs de la colonne VY correspondent aux tangentes hyperboliques du contenu de la colonne V3. Toutes les valeurs sont en degrés.
$TEMP = \text{TANH}(V1[1:5]:V2)$ Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 5 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux tangentes hyperboliques des lignes correspondantes de la colonne V1 et les valeurs de la colonne VX correspondent aux tangentes hyperboliques des lignes correspondantes de la colonne V2. Toutes les valeurs sont en radians.

## Fonctions associées

Fonction	Description
ATAN	Calcule l'arc tangente du contenu de la plage de données spécifiée
COSH	Calcule le cosinus hyperbolique du contenu de la plage de données spécifiée
COT	Calcule la cotangente du contenu de la plage de données spécifiée
SINH	Calcule le sinus hyperbolique du contenu de la plage de données spécifiée
TAN	Calcule la tangente du contenu de la plage de données spécifiée

---

## Macro TOTAL

est disponible dans IBM Campaign et IBM Interact. TOTAL

### Syntaxe

TOTAL(data [, keyword])

### Paramètres

data

Valeurs numériques dont la somme doit être calculée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie

comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

keyword

Ce mot clé facultatif détermine la manière dont le calcul est effectué sur la plage de données d'entrée. Sélectionnez l'une des valeurs suivantes :

ALL - Effectue le calcul sur toutes les cibles de data (valeur par défaut)

COL - Effectue le calcul séparément pour chaque colonne de data

ROW - Effectue le calcul séparément pour chaque ligne de data

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

**Remarque :** De nombreuses fonctions de macro utilisent les paramètres de mot clé {ALL | COL | ROW}. Ces mots clés ne s'appliquent pas dans IBM Campaign car les données d'entrées correspondent toujours à une colonne ou à une zone. La macro se comporte toujours comme si le mot clé COL a été spécifié. Par conséquent, vous n'avez pas besoin de spécifier ces mots clés lorsque vous utilisez IBM Campaign.

## Description

TOTAL calcule la somme de toutes les cibles de la plage de données spécifiée.

**Remarque :** TOTAL est identique à la fonction macro SUM.

## Exemples

TEMP = TOTAL(3) Crée une colonne nommée TEMP contenant la valeur trois.
TEMP = TOTAL((COLUMN(3, 5, 1))) Crée une colonne nommée TEMP contenant la valeur neuf.
TEMP = TOTAL(V1) Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la somme du contenu de la colonne V1.
TEMP = TOTAL(V1:V3) Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la somme du contenu des colonnes V1, V2 et V3.
TEMP = TOTAL(V1[1:5]:V4) Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la somme des cibles des lignes 10 à 20 des colonnes V1 à V4.
TEMP = TOTAL(V1:V3, COL) Crée trois colonnes nommées TEMP, VX et VY. La valeur unique de la colonne TEMP correspond à la somme du contenu de la colonne V1, la valeur unique de la colonne VX correspond à la somme du contenu de la colonne V2 et la valeur unique de la colonne VY correspond à la somme du contenu de la colonne V3.

<p>TEMP = TOTAL(V1[1:5]:V3, COL)</p> <p>Crée trois colonnes nommées TEMP, VX et VY contenant chacune une seule valeur. La valeur de la colonne TEMP correspond à la somme des cibles des lignes 1 à 5 de la colonne V1, la valeur de la colonne VX correspond à la somme des cibles des lignes 1 à 5 de la colonne V2, et la valeur de la colonne VY correspond à la somme des cibles des lignes 1 à 5 de la colonne V3.</p>
<p>TEMP = TOTAL(V1:V3, ROW)</p> <p>Crée une colonne nommée TEMP dans laquelle chaque entrée de cible correspond à la somme de la ligne correspondante dans les colonnes V1, V2 et V3.</p>
<p>TEMP = TOTAL(V1[1:5]:V3, ROW)</p> <p>Crée une colonne nommée TEMP, dans laquelle les cibles des lignes 1 à 5 contiennent la somme de la ligne correspondante dans les colonnes V1 à V3. Les autres cibles de TEMP sont vides.</p>

## Fonctions associées

Fonction	Description
AVG ou MEAN	Calcule la moyenne arithmétique d'une plage de cibles

---

## Macro TRUNCATE

est disponible uniquement dans IBM Campaign. TRUNCATE

### Syntaxe

TRUNCATE(data)

### Paramètres

data

Valeurs numériques à tronquer. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

TRUNCATE calcule la partie entière de chaque valeur dans la plage de données spécifiée. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant la partie entière (sans partie décimale) des nombres dans la colonne d'entrée correspondante.

**Remarque :** Les fonctions macro FRACTION et TRUNCATE sont complémentaires car leur somme correspond aux valeurs d'origine.



## Exemples

TEMP = TRUNCATE(4.3) Crée une colonne nommée TEMP contenant la valeur 4.
TEMP = TRUNCATE(2.9) Crée une colonne nommée TEMP contenant la valeur -2.
TEMP = TRUNCATE(V1) Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond à la partie décimale du contenu de la colonne V1.
TEMP = TRUNCATE(V1:V3) Crée trois colonnes nommées TEMP, VX et VY. Les valeurs de la colonne TEMP correspondent aux parties tronquées de la colonne V1, les valeurs de la colonne VX correspondent aux parties tronquées de la colonne V2 et les valeurs de la colonne VY correspondent aux parties tronquées de la colonne V3.
TEMP = TRUNCATE(V1[10:20]) Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent les parties tronquées des valeurs des lignes 10 à 20 de la colonne V1. Les autres cibles de TEMP sont vides.
TEMP = TRUNCATE(V1[50:99]:V2) Crée deux colonnes nommées TEMP et VX, contenant des valeurs dans les lignes 1 à 50 (les autres cibles sont vides). Les valeurs de la colonne TEMP correspondent aux parties tronquées des lignes de la colonne V1 et les valeurs de la colonne VX correspondent aux parties tronquées des valeurs de la colonne V2.

## Fonctions associées

Fonction	Description
CEILING	Calcule le plafond de chaque valeur de la plage de données spécifiée
FLOOR	Calcule le plancher de chaque valeur de la plage de données spécifiée
FRACTION	Renvoie la partie décimale de chaque valeur de la plage de données spécifiée

---

## Macro UPPER

est disponible dans IBM Campaign et IBM Interact. UPPER

### Syntaxe

UPPER(data)

### Paramètres

data

Valeur de chaîne à convertir en majuscules.

## Description

UPPER convertit chaque valeur de chaîne de la plage de données spécifiée en majuscules. Il renvoie une nouvelle colonne par colonne d'entrée, chacune contenant la chaîne en majuscules dans la colonne d'entrée correspondante.

## Exemples

Temp = UPPER "gold"
Crée une colonne nommée Temp contenant "GOLD".
TEMP = UPPER( "jan 15, 1997")
Crée une colonne nommée TEMP, qui contient la chaîne de texte ASCII " JAN 15, 1997 ".
TEMP = UPPER( "Pressure")
Crée une colonne nommée TEMP, qui contient la chaîne de texte ASCII "PRESSURE".
TEMP = UPPER(V1)
Crée une colonne nommée TEMP contenant les caractères en majuscules de chaque chaîne de la colonne V1.

---

## Macro VARIANCE

est disponible uniquement dans IBM Campaign. VARIANCE

### Syntaxe

VARIANCE(data [, keyword])

### Paramètres

data

Valeurs numériques dont la variance doit être calculée. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

keyword

Ce mot clé facultatif détermine la manière dont le calcul est effectué sur la plage de données d'entrée. Sélectionnez l'une des valeurs suivantes :

ALL - Effectue le calcul sur toutes les cibles de data (valeur par défaut)

COL - Effectue le calcul séparément pour chaque colonne de data

ROW - Effectue le calcul séparément pour chaque ligne de data

Pour plus de détails sur l'utilisation des mots clés dans IBM Campaign, voir «Spécifications de format», à la page 9.

**Remarque :** De nombreuses fonctions macro acceptent les paramètres de mot clé {ALL | COL | ROW}. Ces mots clés ne s'appliquent pas dans **IBM Campaign** car les

données d'entrées correspondent toujours à une colonne ou à une zone. La macro se comporte toujours comme si le mot clé COL a été spécifié. Par conséquent, vous n'avez pas besoin de spécifier ces mots clés lorsque vous utilisez **IBM Campaign**.

## Description

VARIANCE calcule la variance de toutes les valeurs de la plage de données spécifiée. La variance est égale au carré de l'écart type. La variance est calculée comme suit :

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

où les  $x$  représentent les échantillons,  $n$  représente le nombre d'échantillons et  $\text{mean}$  est la moyenne de la distribution.

**Remarque :** Si le nombre d'échantillons  $n = 1$ , VARIANCE renvoie une erreur.

## Exemples

<p>TEMP = VARIANCE(V1)</p> <p>Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la variance du contenu de la colonne V1.</p>
<p>TEMP = VARIANCE(V1:V3)</p> <p>Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la variance du contenu des colonnes V1, V2 et V3.</p>
<p>TEMP = VARIANCE(V1[10:20])</p> <p>Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la variance des cibles des lignes 10 à 20 de la colonne V1.</p>
<p>TEMP = VARIANCE(V1[1:5]:V4)</p> <p>Crée une colonne nommée TEMP contenant une valeur unique qui correspond à la variance des cibles des lignes 1 à 5 des colonnes V1 à V4.</p>
<p>TEMP = VARIANCE(V1:V3, COL)</p> <p>Crée trois colonnes nommées TEMP, VX et VY. La valeur unique de la colonne TEMP correspond à la variance du contenu de la colonne V1, la valeur unique de la colonne VX correspond à la variance du contenu de la colonne V2 et la valeur unique de la colonne VY correspond à la variance du contenu de la colonne V3.</p>
<p>TEMP = VARIANCE_(V1[1:5]:V3, COL) ou TEMP = VARIANCE(V1[1:5]:V3[1:5], COL)</p> <p>Crée trois colonnes nommées TEMP, VX et VY contenant chacune une seule valeur. La valeur de la colonne TEMP correspond à la variance des cibles des lignes 1 à 5 de la colonne V1, la valeur de la colonne VX correspond à la variance des cibles des lignes 1 à 5 de la colonne V2, et la valeur de la colonne VY correspond à la variance des cibles des lignes 1 à 5 de la colonne V3.</p>
<p>TEMP = VARIANCE(V1:V3, ROW)</p> <p>Crée une colonne nommée TEMP dans laquelle chaque entrée de cible correspond à la variance de la ligne correspondante dans les colonnes V1, V2 et V3.</p>

```
TEMP = VARIANCE(V1[1:5]:V3,ROW) ou TEMP = VARIANCE(V1[1:5]:V3[1:5], ROW)
```

Crée une colonne nommée TEMP, dans laquelle les cibles des lignes 1 à 5 contiennent la variance de la ligne correspondante dans les colonnes V1 à V3. Les autres cibles de TEMP sont vides.

---

## Macro WEEKDAY

La macro WEEKDAY est disponible uniquement dans IBM Campaign.

### Syntaxe

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

### Paramètres

data

Dates de texte ASCII à convertir en valeurs numériques représentant les jours de la semaine (1-7). Il peut s'agir de texte ASCII entre guillemets, d'une colonne de texte, d'une plage de cibles contenant du texte ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

conversion\_keyword

Ce mot clé facultatif indique comment interpréter les formats de texte des dates et des heures. Sélectionnez l'une des valeurs suivantes :

1 - mm/dd/yy (valeur par défaut)

2 - dd-mmm-yy

3 - mm/dd/yy hh:mm

Si ce paramètre n'est pas spécifié, la valeur par défaut est 1.

### Description

WEEKDAY convertit les valeurs de texte dans la plage de données spécifiée en valeurs numériques représentant les jours de la semaine, à l'aide du format spécifié pour convertir les dates et les heures. Le nombre 0 correspond à dimanche, 1 à lundi, et ainsi de suite jusqu'à 6 pour samedi. Si une chaîne de texte ne peut pas être analysée à l'aide du mot clé conversion\_keyword spécifié, WEEKDAY renvoie une erreur.

### Exemples

```
TEMP = WEEKDAY("1/1/95")
```

Crée une colonne nommée TEMP contenant le nombre 0 (1er janvier 1995 est un dimanche).

<p>TEMP = WEEKDAY(V1, 2)</p> <p>Crée une colonne nommée TEMP contenant les nombres des jours de la semaine pour les chaînes de texte de la colonne V1. Toutes les chaînes de texte de la colonne V1 doivent être au format dd-mmm-yy (sinon des ??? sont renvoyés).</p>
<p>TEMP = WEEKDAY(V1:V3, 3)</p> <p>Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient des nombres représentant les jours de la semaine des chaînes de texte de la colonne V1. La colonne VX contient des nombres représentant les jours de la semaine des chaînes de texte de la colonne V2. La colonne VY contient des nombres représentant les jours de la semaine des chaînes de texte de la colonne V3. Toutes les chaînes de texte des colonnes V1 à V3 doivent se présenter sous la forme mm/dd/yy hh:mm (sinon, des ??? sont renvoyés).</p>
<p>TEMP = WEEKDAY(V1[10:20]:V2, 10)</p> <p>Crée deux colonnes nommées TEMP et VX. La colonne TEMP contient des nombres représentant les jours de la semaine des chaînes de texte des lignes 10 à 20 de la colonne V1. La colonne VX contient des nombres représentant les jours de la semaine des chaînes de texte des lignes 10 à 20 de la colonne V2. Toutes les chaînes de texte doivent se présenter sous la forme mm/dd/yy (sinon, des ??? sont renvoyés).</p>

## Fonctions associées

Fonction	Description
NUMBER	Convertit les chaînes de texte ASCII des heures et des dates en valeurs numériques

---

## Macro WEEKDAYOF

La macro WEEKDAYOF est disponible uniquement dans IBM Campaign.

### Syntaxe

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

### Paramètres

date\_string

Texte représentant une date valide.

input\_format

Un des mots clés du tableau ci-après, spécifiant le format de date de date\_string.

### Description

WEEKDAYOF renvoie le jour de la semaine sous la forme d'un nombre compris entre 0 et 6 (dimanche = 0, lundi = 1, et ainsi de suite) pour la date indiquée par date\_string. Si input\_format n'est pas fourni, le mot clé par défaut DELIM\_M\_D\_Y est utilisé.

## Exemples

WEEKDAYOF("08312000", MMDDYYYY) renvoie le nombre 4 car jeudi est le quatrième jour de la semaine.

**Remarque :** Pour des informations supplémentaires sur les formats de date valides, voir «DATE», à la page 52.

## Fonctions associées

Fonction	Description
DAYOF	Renvoie le jour du mois sous la forme d'un nombre.
MONTHOF	Renvoie le mois de l'année sous la forme d'un nombre.
YEAROF	Renvoie l'année sous la forme d'un nombre.

---

## Macro XOR

La macro XOR est disponible dans IBM Campaign et IBM Interact.

### Syntaxe

```
data1 XOR data2
```

### Paramètres

data1

Entiers non négatifs dont le XOR bit à bit doit être calculé avec les valeurs de data2. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

data2

Entiers non négatifs dont le XOR bit à bit doit être calculé avec les valeurs de data1. Il peut s'agir d'une valeur constante, d'une colonne, d'une plage de cibles ou d'une expression convertie comme l'une de ces valeurs. Le nombre de colonnes dans data2 doit correspondre au nombre de colonnes dans data1, à moins que data2 ne soit une constante. Pour la définition du format de data, voir la section "Paramètres des fonctions macro" du chapitre du présent document correspondant à votre produit IBM.

### Description

XOR calcule le XOR bit à bit entre les deux plages de données spécifiées. Il renvoie une nouvelle colonne pour chaque colonne d'entrée, chacune contenant la colonne correspondante de data1 dont le XOR bit à bit avec la colonne correspondante de data2 a été calculé (à savoir, le XOR bit à bit de la première colonne de data1 avec la première colonne de data, celui de la deuxième colonne avec la deuxième colonne et ainsi de suite).

Si data2 est une constante, le XOR bit à bit de chaque valeur de data1 avec cette valeur est calculé. Si data2 contient une ou plusieurs colonnes, les calculs sont effectués ligne par ligne entre une colonne de data1 et une colonne de data2. Le XOR bit à bit de la première ligne de data1 avec la première ligne de data2 est calculé, la seconde ligne avec la seconde ligne, etc. Ce calcul ligne par ligne génère un résultat pour chaque ligne jusqu'à la dernière valeur de la colonne la plus courte.

**Remarque :** La précision de cette fonction macro est limitée aux valeurs entières inférieures à  $2^{24}$ . Les valeurs négative sont interdites.

## Exemples

TEMP = 3 XOR 7
Crée une colonne nommée TEMP contenant la valeur quatre (le XOR bit à bit de 011 et 111 est égal à 100).
TEMP = V1 XOR 8
Crée une colonne nommée TEMP, dans laquelle chaque valeur correspond au XOR bit à bit du contenu de la colonne V1 avec la valeur binaire 1000.
TEMP = V1 XOR V1
Crée une colonne nommée TEMP ne contenant que des zéros (un XOR entre toute valeur et elle-même génère zéro).
TEMP = V1 XOR V2
Crée une colonne nommée TEMP, dans laquelle chaque valeur est la valeur de ligne de la colonne V1 dont le XOR bit à bit est calculé avec la valeur de ligne correspondante de la colonne V2.
TEMP = V1:V3 XOR V4:V6
Crée trois colonnes nommées TEMP, VX et VY. La colonne TEMP contient les valeurs de V1 dont le XOR bit à bit avec les valeurs de ligne correspondantes de la colonne V4 a été calculé. La colonne VX contient les valeurs résultant du XOR bit à bit des colonnes V2 et V5. La colonne VY contient les valeurs résultant du XOR bit à bit des colonnes V3 et V6.
TEMP = V1[10:20] XOR V2 ou TEMP = V1[10:20] XOR V2[1:11]
Crée une colonne nommée TEMP, dans laquelle les 11 premières cibles contiennent le résultat du XOR bit à bit des valeurs des lignes 10 à 20 de la colonne V1 avec les valeurs des lignes 1 à 11 de la colonne V2. Les autres cibles de TEMP sont vides.

## Fonctions associées

Fonction	Description
BIT_AND	Calcule le AND bit à bit entre deux plages de données spécifiées
BIT_NOT	Calcule le NOT bit à bit du contenu de la plage de données spécifiée
BIT_OR	Calcule le OR bit à bit entre deux plages de données spécifiées

---

## Macro YEAROF

La macro YEAROF est disponible uniquement dans IBM Campaign.

## Syntaxe

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

## Paramètres

date\_string

Texte représentant une date valide.

input\_format

Un des mots clés du tableau ci-après, spécifiant le format de date de date\_string.

## Description

YEAROF renvoie l'année sous la forme d'un nombre pour la date indiquée par date\_string. Si input\_format n'est pas fourni, le mot clé par défaut DELIM\_M\_D\_Y est utilisé.

## Exemples

YEAROF("31082000", DDMMYYYY) renvoie le nombre 2000.

Pour des informations supplémentaires sur les formats de date valides, voir «DATE», à la page 52.

## Fonctions associées

Fonction	Description
DAYOF	Renvoie le jour du mois sous la forme d'un nombre.
MONTHOF	Renvoie le mois de l'année sous la forme d'un nombre.
WEEKDAYOF	Renvoie le jour de la semaine sous la forme d'un nombre.



---

## Comment contacter le support technique IBM

Si vous rencontrez un problème que vous ne parvenez pas à résoudre en consultant la documentation, le correspondant désigné pour le support technique de votre entreprise peut contacter le support technique d'IBM. Pour permettre une résolution efficace et rapide du problème, réunissez les informations nécessaires avant de passer votre appel.

Si vous n'êtes pas le correspondant désigné pour le support technique dans votre société, contactez l'administrateur IBM pour plus d'informations.

### Informations à réunir

Avant de contacter le support technique d'IBM, rassemblez les informations suivantes :

- Une brève description de la nature du problème rencontré
- Les messages d'erreur détaillés s'affichant lorsque le problème se produit
- La liste des étapes complètes permettant de reproduire l'erreur.
- Les fichiers journaux, fichiers de session, fichiers de configuration et fichiers de données connexes
- Les informations sur l'environnement de votre système et de votre produit, que vous pouvez obtenir en procédant comme indiqué dans la section "Informations sur le système".

### Informations sur le système

Lorsque vous appellerez le support technique d'IBM, vous devrez sans doute fournir des informations relatives à votre environnement.

Si le problème rencontré ne vous empêche pas de vous connecter, vous trouverez la plupart de ces informations sur la page A propos de qui fournit des informations sur les applications IBM.

Vous pouvez accéder à la page A propos de en sélectionnant **Aide > A propos de**. Si la page A propos de n'est pas accessible, vous pouvez obtenir le numéro de version d'une application IBM en affichant le fichier `version.txt` qui se trouve dans le répertoire d'installation des différentes applications.

### Informations de contact du support technique d'IBM

Pour savoir comment contacter le support technique IBM, consultez le site Web de support technique IBM : ([http://www.ibm.com/support/entry/portal/open\\_service\\_request](http://www.ibm.com/support/entry/portal/open_service_request)).

**Remarque :** Pour entrer une demande de support, vous devez vous connecter avec un compte IBM. Si possible, ce compte doit être associé à votre numéro client IBM. Pour en savoir plus sur l'association de votre compte à votre numéro de client IBM, accédez à **Ressources de support > ESS (Entitled Software Support)** dans le portail du support.



---

## Remarques

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, programme ou service IBM n'implique pas que seul ce produit, programme ou service IBM puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous accorde aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

Pour le Canada, veuillez adresser votre courrier à :

IBM Director of Commercial Relations  
IBM Canada Ltd  
3600 Steeles Avenue East  
Markham, Ontario  
L3R 9Z7  
Canada

Les informations sur les licences concernant les produits utilisant un jeu de caractères double octet peuvent être obtenues par écrit à l'adresse suivante :

Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan, Ltd.  
19-21, Nihonbashi-Hakozakicho, Chuo-ku  
Tokyo 103-8510, Japon

Le paragraphe suivant ne s'applique ni au Royaume-Uni, ni dans aucun pays dans lequel il serait contraire aux lois locales. LE PRESENT DOCUMENT EST LIVRE EN L'ETAT SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFAÇON AINSI QU'EN CAS DE DEF AUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Corporation  
B1WA LKG1  
550 King Street  
Littleton, MA 01460-1250  
U.S.A.

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans ce document et tous les éléments sous licence disponibles s'y rapportant sont fournis par IBM conformément aux dispositions du Livret contractuel (LC7), des Conditions internationales d'utilisation de logiciels IBM ou de tout autre accord équivalent.

Les données de performance indiquées dans ce document ont été déterminées dans un environnement contrôlé. Par conséquent, les résultats peuvent varier de manière significative selon l'environnement d'exploitation utilisé. Certaines mesures évaluées sur des systèmes en cours de développement ne sont pas garanties sur tous les systèmes disponibles. En outre, elles peuvent résulter d'extrapolations. Les résultats peuvent donc varier. Il incombe aux utilisateurs de ce document de vérifier si ces données sont applicables à leur environnement d'exploitation.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles. IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle ne peut recevoir aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Toute instruction relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être changée ou annulée sans préavis, et doit être considérée uniquement comme un objectif.

Tous les tarifs indiqués sont les prix de vente actuels suggérés par IBM et sont susceptibles d'être changés sans préavis. Les tarifs appliqués peuvent varier selon les revendeurs.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs d'individus, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms d'individus, de sociétés ou des données réelles serait purement fortuite.

#### LICENCE DE COPYRIGHT :

Le présent guide contient des exemples de programmes d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes à l'interface de programme d'application de la plateforme pour lesquels ils ont été écrits. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes. Les exemples de programmes sont fournis en l'état, sans garantie d'aucune sorte. IBM ne pourra en aucun cas être tenue responsable des dommages liés à l'utilisation des exemples de programmes.

Si vous visualisez ces informations en ligne, il se peut que les photographies et illustrations en couleur n'apparaissent pas à l'écran.

---

## Marques

IBM, le logo IBM et [ibm.com](http://ibm.com) sont des marques d'International Business Machines Corp. dans de nombreux pays. Les autres noms de produits et de services peuvent être des marques d'IBM ou d'autres sociétés. La liste actualisée de toutes les marques d'IBM est disponible sur la page Web "Copyright and trademark information" à l'adresse [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

---

## Règles de confidentialité et conditions d'utilisation

Les Logiciels IBM, y compris les Logiciels sous forme de services ("Offres Logiciels") peuvent utiliser des cookies ou d'autres technologies pour collecter des informations sur l'utilisation des produits, améliorer l'acquis utilisateur, personnaliser les interactions avec celui-ci, ou dans d'autres buts. Un cookie est une donnée qu'un site Web peut envoyer à votre navigateur et qui peut ensuite être stockée sur votre ordinateur sous la forme d'une balise identifiant ce dernier. Bien souvent, aucune information personnelle identifiable n'est collectée par les Offres Logiciels. Si la présente Offre Logiciels utilise des cookies pour collecter des informations personnelles identifiables, des informations spécifiques sur cette utilisation sont fournies ci-dessous.

Selon la configuration déployée, la présente Offre Logiciels peut utiliser des cookies de session et des cookies persistants destinés à collecter le nom et le mot de passe des utilisateurs pour les fonctions de gestion des sessions et d'authentification, pour faciliter l'utilisation des produits, ou pour d'autres objectifs de suivi de l'utilisation ou fonctionnels. Ces cookies peuvent être désactivés mais leur désactivation élimine également la fonctionnalité qu'ils activent.

Diverses juridictions régulent la collecte d'informations personnelles via les cookies et autres technologies similaires. Si les configurations déployées de cette Offre Logiciels vous permettent, en tant que client, de collecter des informations permettant d'identifier les utilisateurs par l'intermédiaire de cookies ou par d'autres techniques, vous devez solliciter un avis juridique sur la réglementation applicable à ce type de collecte, notamment en termes d'information et de consentement.

IBM demande à ses clients (1) de fournir un lien clair et visible vers les conditions d'utilisation et la politique de protection des renseignements personnels du site Web du Client, ainsi qu'un lien vers la collecte de données et les pratiques d'utilisation d'IBM et du Client, (2) de signaler que les cookies et les images de pistage (clear gifs/web beacons) sont copiés sur l'ordinateur du visiteur par IBM au nom du Client, et de fournir une explication sur l'objectif et l'utilisation de ces technologies, et (3) selon les conditions requises par la loi, d'obtenir le consentement des visiteurs du site Web avant de placer les cookies et les images de pistage déposés par le Client ou par IBM au nom du Client sur leurs machines.

Pour plus d'informations sur l'utilisation à ces fins des différentes technologies, y compris celle des cookies, consultez les Points principaux de la Déclaration IBM de confidentialité sur Internet à l'adresse <http://www.ibm.com/privacy/details/us/> en dans la section intitulée "Cookies, Web Beacons and Other Technologies."



