

# **Manuel utilisateur de Calligra Sheets**

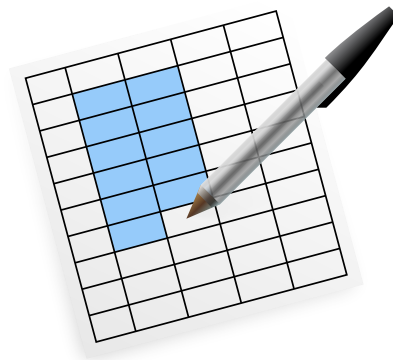
**Pamela Roberts**

**Anne-Marie Mahfouf**

**Gary Cramblitt**

**Traduction française: Gérard Delafond**

**Traduction française: Ludovic Grossard**



## Manuel utilisateur de Calligra Sheets

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>16</b>
<b>2</b>	<b>Bases de Calligra Sheets</b>	<b>17</b>
2.1	Les tableurs pour les débutants . . . . .	17
2.2	Sélection des cellules . . . . .	19
2.3	Saisie des données . . . . .	20
2.3.1	Format de cellule générique . . . . .	20
2.4	Copier, Couper et Coller . . . . .	20
2.4.1	Copier et coller les zones de cellules . . . . .	21
2.4.2	Autres modes de collage . . . . .	21
2.5	Insérer et supprimer . . . . .	22
2.6	Sommes simples . . . . .	22
2.6.1	Re-calcul . . . . .	23
2.7	Tri des données . . . . .	23
2.8	La calculette de sommaire de la barre d'état . . . . .	24
2.9	Enregistrer votre travail . . . . .	24
2.9.1	Modèles . . . . .	25
2.10	Imprimer un tableau . . . . .	25
<b>3</b>	<b>Formatage du Tableur</b>	<b>27</b>
3.1	Format des cellules . . . . .	27
3.1.1	Formats de données et représentation . . . . .	28
3.1.2	Réglages des polices et du texte . . . . .	30
3.1.3	Position et rotation du texte . . . . .	31
3.1.4	Bordure de cellule . . . . .	33
3.1.5	Arrière-plan de cellule . . . . .	34
3.1.6	Protection de cellule . . . . .	34
3.2	Attributs conditionnels des cellules . . . . .	35
3.3	Changer la taille des cellules . . . . .	35
3.4	Fusionner les cellules . . . . .	36
3.5	Cacher lignes et colonnes . . . . .	36
3.6	Propriétés de la feuille . . . . .	36

<b>4</b>	<b>Calligra Sheets expert</b>	<b>39</b>
4.1	Séries	39
4.2	Formules	40
4.2.1	Fonctions intégrées	40
4.2.2	Comparaisons logiques	40
4.2.3	Références absolues aux cellules	41
4.3	Arithmétique en utilisant le collage spécial	41
4.4	Formules de tableau	42
4.5	Recherche de solution	42
4.6	Tables avec pivot	42
4.7	Utiliser plusieurs feuilles de travail	44
4.7.1	Consolider des données	44
4.8	Insérer un diagramme	45
4.9	Insérer des données externes	46
4.10	Lier les cellules	47
4.11	Vérification de validité	47
4.12	Protection	47
4.12.1	Protection de document	47
4.12.2	Protection de la feuille	49
4.12.3	Protection d'une cellule ou d'une sélection de cellules	49
4.12.4	Masquer la formule de la cellule	50
4.12.5	Masquer tout dans la cellule	51
4.13	Autres fonctionnalités	51
4.13.1	Zones et cellules nommées	51
4.13.2	Commentaires de cellules	52
<b>5</b>	<b>Configurer les raccourcis et barres d'outils de Calligra Sheets</b>	<b>53</b>
5.1	Raccourcis	53
5.2	Barres d'outils	54
<b>6</b>	<b>La boîte de dialogue de configuration de Calligra Sheets</b>	<b>55</b>
6.1	Interface	55
6.2	Ouvrir / Enregistrer	57
6.3	Modules externes	58
6.4	Vérification d'orthographe	58
6.5	Auteur	59
<b>7</b>	<b>Références des commandes</b>	<b>60</b>
7.1	Le menu Fichier	60
7.2	Le menu Modifier	61
7.3	Le menu Vue	62
7.4	Le menu Aller	62
7.5	Le menu Insertion	62
7.6	Le menu Format	62
7.7	Le menu Données	63
7.8	Le menu Outils	64
7.9	Le menu Configuration	64
7.10	Le menu Aide	65
7.11	Le menu du bouton droit de la souris	65
7.12	Autres raccourcis	67

<b>8 Fonctions</b>	<b>68</b>
8.1 Fonctions prises en charge	68
8.1.1 Opérations par bit	68
8.1.1.1 BITAND	68
8.1.1.2 BITLSHIFT	69
8.1.1.3 BITOR	69
8.1.1.4 BITRSHIFT	69
8.1.1.5 BITXOR	70
8.1.2 Conversion	70
8.1.2.1 ARABIC	70
8.1.2.2 ASCIITOCHAR	70
8.1.2.3 BOOL2INT	71
8.1.2.4 BOOL2STRING	71
8.1.2.5 CARX	71
8.1.2.6 CARY	72
8.1.2.7 CHARTOASCII	72
8.1.2.8 DECSEX	72
8.1.2.9 INT2BOOL	73
8.1.2.10 NUM2STRING	73
8.1.2.11 POLA	73
8.1.2.12 POLR	74
8.1.2.13 ROMAN	74
8.1.2.14 SEXDEC	75
8.1.2.15 CHAÎNE	75
8.1.3 Base de données	75
8.1.3.1 DAVERAGE	75
8.1.3.2 DCOUNT	76
8.1.3.3 DCOUNTA	76
8.1.3.4 DGET	76
8.1.3.5 DMAX	77
8.1.3.6 DMIN	77
8.1.3.7 DPRODUCT	77
8.1.3.8 DSTDEV	78
8.1.3.9 DSTDEVP	78
8.1.3.10 DSUM	78
8.1.3.11 DVAR	79
8.1.3.12 DVARP	79
8.1.3.13 GETPIVOTDATA	79
8.1.4 Date et heure	80
8.1.4.1 CURRENTDATE	80
8.1.4.2 CURRENTDATETIME	80
8.1.4.3 CURRENTTIME	80
8.1.4.4 DATE	80
8.1.4.5 DATE2UNIX	81
8.1.4.6 DATEDIF	81
8.1.4.7 DATEVALUE	81

## Manuel utilisateur de Calligra Sheets

8.1.4.8	DAY	82
8.1.4.9	DAYNAME	82
8.1.4.10	DAYOFYEAR	82
8.1.4.11	DAYS	83
8.1.4.12	DAYS360	83
8.1.4.13	DAYSINMONTH	83
8.1.4.14	DAYSINYEAR	84
8.1.4.15	EASTERSUNDAY	84
8.1.4.16	EDATE	84
8.1.4.17	EOMONTH	85
8.1.4.18	HOUR	85
8.1.4.19	HOURS	85
8.1.4.20	ISLEAPYEAR	86
8.1.4.21	ISOWEEKNUM	86
8.1.4.22	MINUTE	86
8.1.4.23	MINUTES	87
8.1.4.24	MONTH	87
8.1.4.25	MONTHNAME	87
8.1.4.26	MONTHS	87
8.1.4.27	NETWORKDAY	88
8.1.4.28	NOW	88
8.1.4.29	SECOND	89
8.1.4.30	SECONDS	89
8.1.4.31	TIME	89
8.1.4.32	TIMEVALUE	90
8.1.4.33	TODAY	90
8.1.4.34	UNIX2DATE	90
8.1.4.35	WEEKDAY	90
8.1.4.36	WEEKNUM	91
8.1.4.37	WEEKS	91
8.1.4.38	WEEKSINYEAR	92
8.1.4.39	WORKDAY	92
8.1.4.40	YEAR	92
8.1.4.41	YEARFRAC	93
8.1.4.42	YEARS	93
8.1.5	Technique	93
8.1.5.1	BASE	93
8.1.5.2	BESSELI	94
8.1.5.3	BESSELJ	94
8.1.5.4	BESSELK	94
8.1.5.5	BESSELY	95
8.1.5.6	BIN2DEC	95
8.1.5.7	BIN2HEX	95
8.1.5.8	BIN2OCT	96
8.1.5.9	COMPLEX	96
8.1.5.10	CONVERT	96

## Manuel utilisateur de Calligra Sheets

8.1.5.11	DEC2BIN	97
8.1.5.12	DEC2HEX	97
8.1.5.13	DEC2OCT	98
8.1.5.14	DELTA	98
8.1.5.15	ERF	98
8.1.5.16	ERFC	99
8.1.5.17	GESTEP	99
8.1.5.18	HEX2BIN	100
8.1.5.19	HEX2DEC	100
8.1.5.20	HEX2OCT	100
8.1.5.21	IMABS	100
8.1.5.22	IMAGINARY	101
8.1.5.23	IMARGUMENT	101
8.1.5.24	IMCONJUGATE	101
8.1.5.25	IMCOS	102
8.1.5.26	IMCOSH	102
8.1.5.27	IMCOT	102
8.1.5.28	IMCSC	102
8.1.5.29	IMCSCH	103
8.1.5.30	IMDIV	103
8.1.5.31	IMEXP	103
8.1.5.32	IMLN	104
8.1.5.33	IMLOG10	104
8.1.5.34	IMLOG2	104
8.1.5.35	IMPOWER	104
8.1.5.36	IMPRODUCT	105
8.1.5.37	IMREAL	105
8.1.5.38	IMSEC	105
8.1.5.39	IMSECH	105
8.1.5.40	IMSIN	106
8.1.5.41	IMSINH	106
8.1.5.42	IMSQRT	106
8.1.5.43	IMSUB	106
8.1.5.44	IMSUM	107
8.1.5.45	IMTAN	107
8.1.5.46	IMTANH	107
8.1.5.47	OCT2BIN	108
8.1.5.48	OCT2DEC	108
8.1.5.49	OCT2HEX	108
8.1.6	Financières	109
8.1.6.1	ACCRINT	109
8.1.6.2	ACCRINTM	109
8.1.6.3	AMORDEGRC	109
8.1.6.4	AMORLINC	110
8.1.6.5	COMPOUND	110
8.1.6.6	CONTINUOUS	111

## Manuel utilisateur de Calligra Sheets

8.1.6.7	COUPNUM	111
8.1.6.8	CUMIPMT	111
8.1.6.9	CUMPRINC	112
8.1.6.10	DB	112
8.1.6.11	DDB	112
8.1.6.12	DISC	113
8.1.6.13	DOLLARDE	113
8.1.6.14	DOLLARFR	114
8.1.6.15	DURATION	114
8.1.6.16	DURATION_ADD	114
8.1.6.17	EFFECT	115
8.1.6.18	EFFECTIVE	115
8.1.6.19	EURO	115
8.1.6.20	EUROCONVERT	116
8.1.6.21	FV	116
8.1.6.22	FV_ANNUITY	116
8.1.6.23	INTRATE	117
8.1.6.24	IPMT	117
8.1.6.25	IRR	118
8.1.6.26	ISPMT	118
8.1.6.27	LEVEL_COUPON	119
8.1.6.28	MDURATION	119
8.1.6.29	MIRR	119
8.1.6.30	NOMINAL	120
8.1.6.31	NPER	120
8.1.6.32	NPV	121
8.1.6.33	ODDLPRICE	121
8.1.6.34	ODDLYIELD	121
8.1.6.35	PMT	122
8.1.6.36	PPMT	122
8.1.6.37	PRICEMAT	123
8.1.6.38	PV	124
8.1.6.39	PV_ANNUITY	124
8.1.6.40	RATE	124
8.1.6.41	RECEIVED	125
8.1.6.42	RRI	125
8.1.6.43	SLN	125
8.1.6.44	SYD	126
8.1.6.45	TBILLEQ	126
8.1.6.46	TBILLPRICE	127
8.1.6.47	TBILLYIELD	127
8.1.6.48	VDB	127
8.1.6.49	XIRR	128
8.1.6.50	XNPV	128
8.1.6.51	YIELDDISC	128
8.1.6.52	YIELDMAT	129



## Manuel utilisateur de Calligra Sheets

8.1.6.53	ZERO_COUPON	129
8.1.7	Information	129
8.1.7.1	ERRORTYPE	129
8.1.7.2	FILENAME	130
8.1.7.3	FORMULA	130
8.1.7.4	INFO	130
8.1.7.5	ISBLANK	130
8.1.7.6	ISDATE	131
8.1.7.7	ISERR	131
8.1.7.8	ISERROR	131
8.1.7.9	ISEVEN	131
8.1.7.10	ISFORMULA	132
8.1.7.11	ISLOGICAL	132
8.1.7.12	ISNA	132
8.1.7.13	ISNONTEXT	132
8.1.7.14	ISNOTTEXT	133
8.1.7.15	ISNUM	133
8.1.7.16	ISNUMBER	133
8.1.7.17	ISODD	134
8.1.7.18	ISREF	134
8.1.7.19	ISTEXT	134
8.1.7.20	ISTIME	135
8.1.7.21	N	135
8.1.7.22	NA	135
8.1.7.23	TYPE	135
8.1.8	Logiques	136
8.1.8.1	AND	136
8.1.8.2	FALSE	136
8.1.8.3	IF	136
8.1.8.4	IFERROR	137
8.1.8.5	IFNA	137
8.1.8.6	NAND	137
8.1.8.7	NOR	138
8.1.8.8	NOT	138
8.1.8.9	OR	138
8.1.8.10	TRUE	139
8.1.8.11	XOR	139
8.1.9	Recherche et références	139
8.1.9.1	ADDRESS	139
8.1.9.2	AREAS	140
8.1.9.3	CELL	140
8.1.9.4	CHOOSE	141
8.1.9.5	COLUMN	141
8.1.9.6	COLUMNS	141
8.1.9.7	HLOOKUP	142
8.1.9.8	INDEX	142

## Manuel utilisateur de Calligra Sheets

8.1.9.9	INDIRECT	142
8.1.9.10	LOOKUP	143
8.1.9.11	MATCH	143
8.1.9.12	MULTIPLE.OPERATIONS	143
8.1.9.13	OFFSET	144
8.1.9.14	ROW	144
8.1.9.15	ROWS	144
8.1.9.16	SHEET	145
8.1.9.17	SHEETS	145
8.1.9.18	VLOOKUP	145
8.1.10	Mathématiques	145
8.1.10.1	ABS	145
8.1.10.2	CEIL	146
8.1.10.3	CEILING	146
8.1.10.4	COUNT	147
8.1.10.5	COUNTA	147
8.1.10.6	COUNTBLANK	148
8.1.10.7	COUNTIF	148
8.1.10.8	CUR	148
8.1.10.9	DIV	149
8.1.10.10	EPS	149
8.1.10.11	EVEN	149
8.1.10.12	EXP	150
8.1.10.13	FACT	150
8.1.10.14	FACTDOUBLE	150
8.1.10.15	FIB	151
8.1.10.16	FLOOR	151
8.1.10.17	GAMMA	152
8.1.10.18	GCD	152
8.1.10.19	G_PRODUCT	152
8.1.10.20	INT	153
8.1.10.21	INV	153
8.1.10.22	KPRODUCT	153
8.1.10.23	LCM	154
8.1.10.24	LN	154
8.1.10.25	LOG	154
8.1.10.26	LOG10	155
8.1.10.27	LOG2	155
8.1.10.28	LOGN	156
8.1.10.29	MAX	156
8.1.10.30	MAXA	156
8.1.10.31	MDETERM	157
8.1.10.32	Min	157
8.1.10.33	MINA	158
8.1.10.34	MINVERSE	158
8.1.10.35	MMULT	158

## Manuel utilisateur de Calligra Sheets

8.1.10.36	MOD	159
8.1.10.37	MROUND	159
8.1.10.38	MULTINOMIAL	160
8.1.10.39	MULTIPLY	160
8.1.10.40	MUNIT	160
8.1.10.41	ODD	161
8.1.10.42	POW	161
8.1.10.43	POWER	161
8.1.10.44	PRODUCT	162
8.1.10.45	QUOTIENT	162
8.1.10.46	RAND	162
8.1.10.47	RANDBERNOULLI	163
8.1.10.48	RANDBETWEEN	163
8.1.10.49	RANDBINOM	163
8.1.10.50	RANDEXP	164
8.1.10.51	RANDNEGBINOM	164
8.1.10.52	RANDNORM	164
8.1.10.53	RANDPOISSON	165
8.1.10.54	ROOTN	165
8.1.10.55	ROUND	165
8.1.10.56	ROUNDDOWN	166
8.1.10.57	ROUNDUP	166
8.1.10.58	SERIESSUM	167
8.1.10.59	SIGN	167
8.1.10.60	SQRT	168
8.1.10.61	SQRTPI	168
8.1.10.62	SUBTOTAL	168
8.1.10.63	SUM	169
8.1.10.64	SUMA	169
8.1.10.65	SUMIF	170
8.1.10.66	SUMSQ	170
8.1.10.67	TRANSPOSE	170
8.1.10.68	TRUNC	171
8.1.11	Statistiques	171
8.1.11.1	AVEDEV	171
8.1.11.2	AVERAGE	172
8.1.11.3	AVERAGEA	172
8.1.11.4	BETADIST	172
8.1.11.5	BETAINV	173
8.1.11.6	BINO	173
8.1.11.7	CHIDIST	174
8.1.11.8	COMBIN	174
8.1.11.9	COMBINA	174
8.1.11.10	CONFIDENCE	175
8.1.11.11	CORREL	175
8.1.11.12	COVAR	175

## Manuel utilisateur de Calligra Sheets

8.1.11.13 DEVSQ . . . . .	176
8.1.11.14 EXPONDIST . . . . .	176
8.1.11.15 FDIST . . . . .	176
8.1.11.16 FINV . . . . .	177
8.1.11.17 FISHER . . . . .	177
8.1.11.18 FISHERINV . . . . .	177
8.1.11.19 FREQUENCY . . . . .	178
8.1.11.20 GAMMADIST . . . . .	178
8.1.11.21 GAMMAINV . . . . .	178
8.1.11.22 GAMMALN . . . . .	179
8.1.11.23 GAUSS . . . . .	179
8.1.11.24 GEOMEAN . . . . .	179
8.1.11.25 HARMEAN . . . . .	180
8.1.11.26 HYPGEOMDIST . . . . .	180
8.1.11.27 INTERCEPT . . . . .	180
8.1.11.28 INVBINO . . . . .	181
8.1.11.29 KURT . . . . .	181
8.1.11.30 KURTP . . . . .	181
8.1.11.31 gros . . . . .	182
8.1.11.32 LEGACYFDIST . . . . .	182
8.1.11.33 LOGINV . . . . .	182
8.1.11.34 LOGNORMDIST . . . . .	183
8.1.11.35 MEDIAN . . . . .	183
8.1.11.36 MODE . . . . .	183
8.1.11.37 NEGBINOMDIST . . . . .	184
8.1.11.38 NORMDIST . . . . .	184
8.1.11.39 NORMINV . . . . .	185
8.1.11.40 NORMSDIST . . . . .	185
8.1.11.41 NORMSINV . . . . .	185
8.1.11.42 PEARSON . . . . .	185
8.1.11.43 PERCENTILE . . . . .	186
8.1.11.44 PERMUT . . . . .	186
8.1.11.45 PERMUTATIONA . . . . .	186
8.1.11.46 PHI . . . . .	187
8.1.11.47 POISSON . . . . .	187
8.1.11.48 RANK . . . . .	187
8.1.11.49 RSQ . . . . .	188
8.1.11.50 SKEW . . . . .	188
8.1.11.51 SKEWP . . . . .	188
8.1.11.52 SLOPE . . . . .	189
8.1.11.53 SMALL . . . . .	189
8.1.11.54 STANDARDIZE . . . . .	189
8.1.11.55 STDEV . . . . .	190
8.1.11.56 STDEVA . . . . .	190
8.1.11.57 STDEVP . . . . .	191
8.1.11.58 STDEVPA . . . . .	191

## Manuel utilisateur de Calligra Sheets

8.1.11.59	STEYX	192
8.1.11.60	SUM2XMY	192
8.1.11.61	SUMPRODUCT	192
8.1.11.62	SUMX2MY2	192
8.1.11.63	SUMX2PY2	193
8.1.11.64	SUMXMY2	193
8.1.11.65	TDIST	193
8.1.11.66	TREND	193
8.1.11.67	TRIMMEAN	194
8.1.11.68	TTEST	194
8.1.11.69	VAR	194
8.1.11.70	VARA	195
8.1.11.71	VARIANCE	195
8.1.11.72	VARP	196
8.1.11.73	VARPA	197
8.1.11.74	WEIBULL	197
8.1.11.75	ZTEST	198
8.1.12	Texte	198
8.1.12.1	ASC	198
8.1.12.2	BAHTTEXT	198
8.1.12.3	CHAR	198
8.1.12.4	CLEAN	199
8.1.12.5	CODE	199
8.1.12.6	COMPARE	199
8.1.12.7	CONCATENATE	200
8.1.12.8	DOLLAR	200
8.1.12.9	EXACT	200
8.1.12.10	FIND	201
8.1.12.11	FINDB	201
8.1.12.12	FIXED	202
8.1.12.13	JIS	202
8.1.12.14	LEFT	202
8.1.12.15	LEFTB	203
8.1.12.16	LEN	203
8.1.12.17	LENB	203
8.1.12.18	LOWER	204
8.1.12.19	MID	204
8.1.12.20	MIDB	204
8.1.12.21	PROPER	205
8.1.12.22	REGEXP	205
8.1.12.23	REGEXPRE	205
8.1.12.24	REPLACE	206
8.1.12.25	REPLACEB	206
8.1.12.26	REPT	206
8.1.12.27	RIGHT	207
8.1.12.28	RIGHTB	207

## Manuel utilisateur de Calligra Sheets

8.1.12.29	ROT13	208
8.1.12.30	SEARCH	208
8.1.12.31	SEARCHB	208
8.1.12.32	SLEEK	209
8.1.12.33	SUBSTITUTE	209
8.1.12.34	T	210
8.1.12.35	TEXT	210
8.1.12.36	TOGGLE	210
8.1.12.37	TRIM	211
8.1.12.38	UNICHAR	211
8.1.12.39	UNICODE	211
8.1.12.40	UPPER	212
8.1.12.41	VALUE	212
8.1.13	Trigonométrie	212
8.1.13.1	ACOS	212
8.1.13.2	ACOSH	213
8.1.13.3	ACOT	213
8.1.13.4	ASIN	213
8.1.13.5	ASINH	214
8.1.13.6	ATAN	214
8.1.13.7	ATAN2	214
8.1.13.8	ATANH	215
8.1.13.9	COS	215
8.1.13.10	COSH	216
8.1.13.11	CSC	216
8.1.13.12	CSCH	216
8.1.13.13	DEGREES	217
8.1.13.14	PI	217
8.1.13.15	RADIANS	217
8.1.13.16	SEC	218
8.1.13.17	SECH	218
8.1.13.18	SIN	218
8.1.13.19	SINH	219
8.1.13.20	TAN	219
8.1.13.21	TANH	219

## 9 Remerciements et licence

220

## **Résumé**

Calligra Sheets est un programme de tableur complet.

# Chapitre 1

## Introduction

Ce manuel est dédié à la mémoire de [Visicalc](#).

**IMPORTANT**

Veillez consulter <http://docs.kde.org> pour les mises à jour de ce document.

Calligra Sheets est un tableur complet. Il fait partie de la suite de productivité Calligra pour KDE. Les autres applications Calligra incluent Calligra Words (traitement de textes), Calligra Stage (créateur de diaporamas) entre autres.

Vous devriez visiter <http://www.kde.org> pour davantage d'informations sur KDE en général, ou le site web de Calligra à <http://www.calligra.org>



## Chapitre 2

# Bases de Calligra Sheets

Pamela Robert

Traduction française : Gérard Delafond

Traduction française : Daniel Durand

Traduction française : Ludovic Grossard

### NOTE

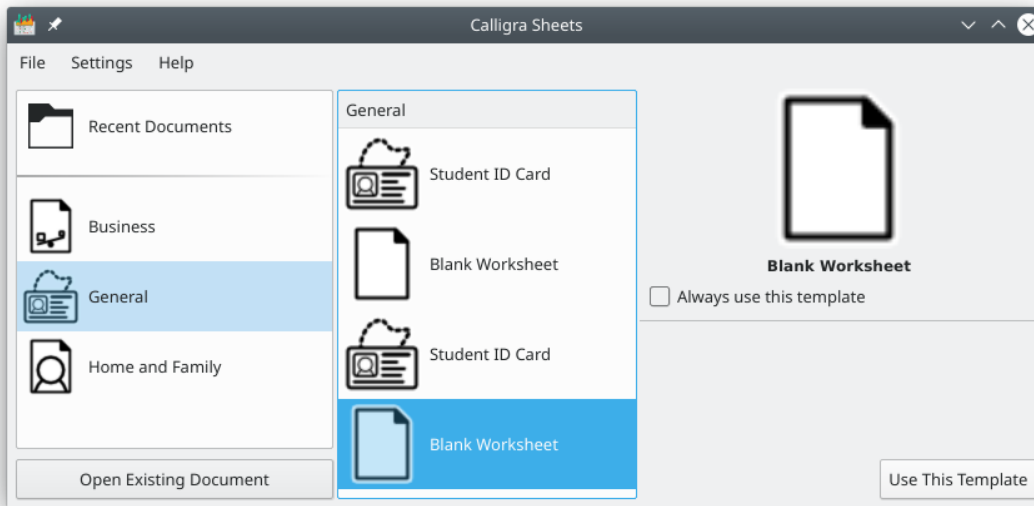
Comme le reste de KDE, Calligra Sheets est hautement configurable, ce qui peut causer des problèmes pour les lecteurs essayant de comparer le texte dans un document comme celui-ci avec ce que voyez sur la version de Calligra Sheets fonctionnant sur leur bureau. Pour couper court à certaines possibilités de confusion, il est suggéré que, quand vous commencez à utiliser Calligra Sheets, vous mettiez les options à leur valeur par défaut dans toutes les pages du dialogue de configuration de Calligra Sheets (obtenu en sélectionnant **Paramètres** → **Configurer Sheets**).

## 2.1 Les tableurs pour les débutants

Cette section tente d'expliquer par exemple ce qu'un tableur comme Calligra Sheets fait en réalité, et pourquoi c'est un outil aussi utile dans de nombreuses situations où vous devez gérer des chiffres. Si vous avez déjà utilisé un tableur, vous pouvez sauter la section suivante.

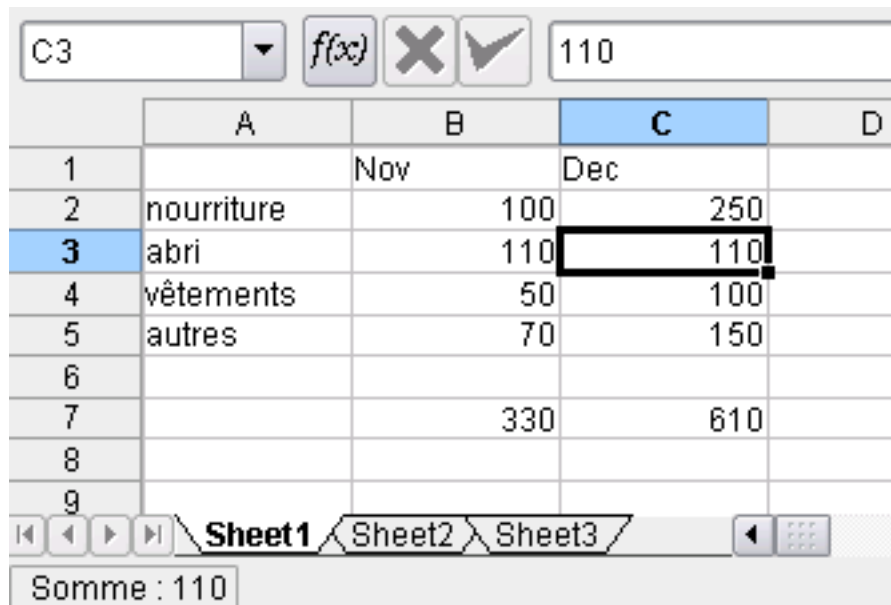
La première chose à faire est de lancer Calligra Sheets. Vous pouvez faire ceci avec le bouton gauche de la souris sur une icône de Calligra Sheets s'il y en a une sur votre bureau ou votre tableau de bord. Ou, vous pouvez sélectionner **Bureautique** → **Calligra Sheets** à partir du lanceur d'applications.

## Manuel utilisateur de Calligra Sheets



Quand il a démarré, on vous demande de créer un document à partir d'un modèle, d'ouvrir un document existant ou un document récent ou de démarrer avec un document vide. Choisir **Commencer avec un document vide**. Puis cliquer sur le bouton **Utiliser ce modèle**.

En regardant Calligra Sheets après son démarrage, vous verrez un tableau de rectangles vides arrangées en lignes numérotées et colonnes lettrées. Cela est l'endroit où vous insérerez des données ou des formules, du texte ou des diagrammes.



	A	B	C	D
1		Nov	Dec	
2	nourriture	100	250	
3	abri	110	110	
4	vêtements	50	100	
5	autres	70	150	
6				
7		330	610	
8				
9				

Maintenant, saisir le texte et les valeurs affichées dans les 5 premières lignes de la capture d'écran ci-dessus dans les cellules correspondantes de votre tableau. Ignorer ce qui se trouve dans la ligne 7 pour le moment. Pour saisir quoi que ce soit dans une cellule, sélectionner d'abord la cellule en cliquant du bouton gauche de la souris dedans, puis écrire ce que vous voulez, puis actionner **Entrée** ou utiliser les touches flèches pour déplacer le point de sélection vers une autre cellule.

Ce que vous avez écrit jusqu'à maintenant pourrait être un simple budget pour les deux mois à venir, une liste de ce que nous avons dépensé pour la nourriture, l'habillement ou toute autre dépense. Maintenant, sélectionnez la cellule B7 (colonne B, ligne 7), saisissez **=B2+B3+B4+B5** et actionnez **Entrée**. Du fait qu'elle commence par le symbole =, Calligra Sheets considère que c'est

une formule, quelque chose qu'il doit calculer, dans ce cas en additionnant ensemble les valeurs dans les 4 cellules B2 à B5, et ce qui sera affiché dans la cellule B7 sera le résultat de ce calcul.

Vous pourriez écrire une formule similaire dans la cellule C7, sauf que dans ce cas, ça aurait dû être  $=C2+C3+C4+C5$ , mais il y a une manière plus simple qui est de copier la cellule B7 et de la coller dans C7. Calligra Sheets ajustera automatiquement les références de cellules de B.. vers C.. quand le collage sera fait.

À ce point, vous pouvez penser que Calligra Sheets ne fait rien de plus que vous ne pourriez faire avec un crayon, un papier et une calculette, et vous auriez raison, mais souvenez-vous que ceci n'est qu'un petit exemple d'un tableur faisant des calculs simples avec peu de nombres. Pour tout tableau de chiffres de taille raisonnable, l'utilisation d'un tableur pour faire des calculs est bien plus rapide et plus précis que de les faire à la main.

De plus, un tableur vous permet de jouer au jeu "Quoi si". Du fait que chaque formule est automatiquement recalculée quand une des valeurs auxquelles elle se rapporte change, vous pouvez voir rapidement ce qui se passe si vous en changez une. En utilisant notre exemple, vous pouvez voir l'effet de réduire le montant dépensé en nourriture en décembre simplement en entrant une nouvelle valeur dans la cellule C2. Si vous aviez un tableur qui modélisait l'effet de serre, vous pourriez peut-être voir l'effet d'une réduction de 50 % de la quantité de méthane relâchée dans l'atmosphère.

## 2.2 Sélection des cellules

Vous pouvez sélectionner une simple cellule ou une zone rectangulaire de cellules dans le tableau. La(les) cellule(s) est(sont) affichée(s) avec une bordure noire.

VOUS POUVEZ SÉLECTIONNER UNE SEULE CELLULE SUIVANT UNE DES MANIÈRES SUIVANTES

- cliquer sur gauche
- Insérer la référence de la cellule (par exemple **B5**) dans la boîte de dialogue de référencement de cellules à gauche de l'outil **Éditeur de cellule** et appuyez sur **Entrée**.
- Utiliser l'option **Aller** → **Aller à la cellule** du menu

Vous pouvez aussi vous déplacer avec les touches **flèches**. En actionnant la touche **Entrée**, la sélection courante se déplacera d'une position vers le haut, vers le bas, vers la gauche ou vers la droite, en fonction des réglages dans la page **Interface** de la **boîte de dialogue de configuration** de Calligra Sheets.

Si vous maintenez enfoncée la touche **Maj** pendant l'utilisation des touches **Flèches**, la sélection se déplacera au début ou à la fin du bloc de cellules occupées.

Pour sélectionner une zone de cellules, faites glisser le pointeur de souris au travers de la zone désirée avec le bouton gauche enfoncé, ou donnez les références des cellules du coin gauche haut et du bas à droite séparées par un deux-points dans la boîte de référence de la cellule de l'**éditeur de cellule** (par exemple **B7 : C14**) et actionnez **Entrée** ou donnez ces références de cellules dans un format similaire dans la boîte de dialogue amenée par **Aller** → **Aller à la cellule...**

Vous pouvez aussi sélectionner une zone de cellules en sélectionnant la cellule dans un coin de la zone désirée, puis tenir la touche **Maj** en utilisant le bouton gauche de la souris pour sélectionner le coin opposé.

Pour sélectionner des cellules non-contiguës, cliquer sur la première cellule que vous souhaitez sélectionner, puis maintenir la touche **Ctrl** enfoncée et sélectionner les autres cellules.

Pour sélectionner une ligne ou une colonne complète de cellules, cliquez du bouton gauche sur le numéro de ligne au bord gauche du tableau ou sur les lettres de colonne en haut. Pour sélectionner des lignes ou des colonnes adjacentes, tirez le pointeur de la souris sur les numéros de ligne ou lettre de colonnes appropriés avec le bouton gauche de la souris enfoncé.

Pour sélectionner des lignes ou des colonnes de cellules non-contiguës, cliquer sur le premier numéro de ligne ou la lettre de la colonne, puis maintenez la touche **Ctrl** enfoncée et sélectionnez les autres lignes ou colonnes de cellules.

## 2.3 Saisie des données

La saisie de données dans une cellule peut être aussi simple que sélectionner des cellules, écrire vos données, puis actionner **Entrée** ou déplacer la sélection vers une autre cellule avec une des touches **flèche**. En fonction de la manière dont vous saisissez les données, Calligra Sheets les interprétera comme un nombre, une date ou de texte :

- Les nombres sont saisis de la manière évidente : **123**, **-123**, **456**, **7** ou en notation scientifique **-1**, **2E-5**.
- Les dates devraient être saisies dans le format de votre “système”, comme défini dans la configuration du système dans l’onglet **Localisation** → **Pays / Région & langue** → **Date et heure**. Si, par exemple, vous utilisez le format JJ/MM/AAAA, vous devez écrire **30/03/2012** pour le 30 mars 2012. Les zéros de début peuvent être omis pour les champs de jour et d’année et seul le dernier ou deux chiffres de l’année doivent être donnés si la date est dans le siècle en cours, par exemple **9/1/9** pour le 9 janvier 2009.
- Les heures doivent aussi être saisies en utilisant le format “Système”. Par exemple, si vous utilisez une horloge sur 12 heures, saisissez les heures en format HH :MM am | pm ou HH :MM :SS am | pm comme **9 : 42 am** ou **10 : 30 : 52 pm**.
- Calligra Sheets définit n’importe quelles données d’entrée comme du “texte” s’il ne peut les reconnaître comme un nombre, une date ou une heure ;

### NOTE

Par défaut, Calligra Sheets justifie à droite les nombres, les dates et les heures dans les cellules et à gauche tout le reste. Ceci peut être un guide utile pour savoir si vous avez saisi une date ou une heure dans le format correct. Mais, souvenez-vous que la manière dont les éléments sont affichés peut se changer en changeant le [format de cellule](#).

La zone de saisie de texte principale dans l’**éditeur de cellule** fournit une manière facile de modifier le contenu de la cellule sélectionnée. Actionner **Entrée** ou cliquer du bouton gauche sur la coche verte quand vous êtes satisfait de ce que vous avez saisi, ou cliquer sur la croix rouge pour annuler vos modifications.

### 2.3.1 Format de cellule générique

Calligra Sheets utilise le format de cellule “Générique” par défaut. Tant que ce format est utilisé, Calligra Sheets détecte automatiquement les données en fonction du format actuel de la cellule. Par exemple, si vous insérez un texte dans une cellule puis saisissez plus tard un nombre dans la même cellule, Calligra Sheets interprétera automatiquement la nouvelle donnée comme un nombre. Si vous souhaitez définir vous-même le type de données, vous devez le paramétrer dans [format de cellule](#). Vous pouvez modifier de nouveau le format vers “Générique” à tout moment.

## 2.4 Copier, Couper et Coller

Au premier coup d’œil, l’élément de menu **Couper**, **Copier** et **Coller** de Calligra Sheets apparaissent similaires à ces fonctions dans les autres applications de KDE. Ayant sélectionné une cellule ou des cellules, vous pouvez choisir **Copier** ou **Couper** dans le menu **Édition** ou dans le menu déroulant que vous obtenez en maintenant enfoncé le bouton droit de la souris sur une cellule sélectionnée. Vous pouvez aussi utiliser les raccourcis clavier **Ctrl+C** ou **Ctrl+X**, puis déplacez la sélection vers la cellule-cible et choisissez **Coller** ou actionnez **Ctrl+V**. Cependant, il y a des subtilités associées à ces fonctions dans Calligra Sheets, qui sont discutées ci-dessous.

Si une cellule contient une formule, la formule elle-même est copiée plutôt que le résultat affiché, et si la formule contient une référence à une autre cellule, cette référence est changée par les

opérations **Couper**, **Copier** et **Coller** au point vers la cellule qui est dans la même position relative à la cellule originale. Par exemple, si la cellule A2 contient la formule **=B3** et est copiée dans C4, la cellule C4 contiendra **=D5**.

Ceci peut sembler une manière plutôt étrange de faire une copie, mais 99 fois sur cent, c'est exactement ce qui est voulu (si ce n'est pas le cas, voyez la section sur [références absolues aux cellules](#)). Par exemple dans la simple liste d'achats ci-dessous, la cellule D2 devrait contenir **=B2 \* C2**, D3 devant être **=B3 \* C3**, D4 devant être **=B4 \* C4** et ainsi de suite. Au lieu d'avoir à saisir une formule différente dans chaque cellule, vous pouvez simplement écrire la première formule dans D2 et la copier dans les cellules dessous, laissant Calligra Sheets ajuster les références aux cellules pour s'y adapter.

	A	B	C	D
1	item		quantité	total
2	pommes (kg)	2,25	1	2,25
3	pain (miche)	0,65	1	0,65
4	lait (L)	0,92	2	1,84
5	nourriture pour chat (conserve)	0,38	5	1,9
6				
7				
8				

## 2.4.1 Copier et coller les zones de cellules

Dans l'exemple ci-dessus, D2 peut être collé dans toutes les trois cellules D3 à D5 d'un coup en copiant simplement D2, puis en sélectionnant la zone complète de cellules D3 :D5 avant de faire le collage.

Une zone rectangulaire de cellules peut être copiée ou coupée d'un coup en sélectionnant la zone avant de faire la copie. Puis, sélectionnez la cellule du coin haut gauche de la zone où vous voulez coller avant de faire le collage.

Si vous coupez ou copiez une zone rectangulaire de cellules, disons B2 :C3 et la collez dans une grande zone plus grande comme A10 :D13, le motif original des cellules sera répétée pour remplir la zone-cible.

Calligra Sheets fournit aussi une méthode de "Tirer et copier" pour copier les cellules dans d'autres cellules immédiatement au-dessous ou à droite des cellules originales. Pour utiliser cette méthode, sélectionnez les cellules à copier, puis positionnez le pointeur de la souris sur le petit carré noir dans le coin en bas à droite des cellules sélectionnées, de telle manière que le curseur se change en flèche à double tête. Puis, tenez le bouton gauche enfoncé pendant que vous tirez les cellules sélectionnées aussi loin que vous voulez. Notez **Maj** que les références de cellules dans les formules sont incrémentées suivant la modification de leur position relative. Les références absolues ne sont pas modifiées.

## 2.4.2 Autres modes de collage

Une cellule peut contenir du texte, une valeur ou une formule, et peut aussi contenir des [informations de formatage](#) de police, bordure ou fond spéciales. Calligra Sheets a des versions spéciales de Coller qui vous permettent de manipuler ces éléments de différentes manières.

**Édition** → **Collage spécial...** ouvre la boîte de dialogue **Collage spécial**. En sélectionnant l'élément approprié de la partie gauche de cette boîte de dialogue, vous pouvez choisir de coller **Tout**, seulement le **Texte**, le **Format** de cellule, tout **Commentaire** dans les cellules ou **Tout sauf les bordures**. Les éléments de la partie basse de cette boîte de dialogue vous permettent de faire de simples **opérations arithmétiques sur une zone de cellules**.

**Coller avec insertion...** insère les cellules copiées dans la feuille en déplaçant les cellules qui autrement seraient écrasées d'un nombre convenable de lignes ou colonnes vers le bas ou la droite. Il peut aussi être utilisé pour insérer des lignes copiées complètes ou des colonnes dans la feuille.

## 2.5 Insérer et supprimer

Utilisez la touche **Suppr** ou **Modifier** → **Effacer** → **Contenu** pour enlever le texte, la valeur ou la formule des cellules, ligne ou colonne sélectionnées sans rien affecter d'autre.

Pour tout supprimer dans les cellules, lignes ou colonnes sélectionnées, y compris les commentaires et le formatage spécial, choisissez l'option **Toutes** du menu **Modifier** → **Effacer** ou depuis le menu contextuel que vous obtenez quand vous cliquez du bouton droit sur une sélection.

Pour enlever les lignes ou colonnes sélectionnées complètement, utilisez les options **Supprimer les lignes** ou **Supprimer les colonnes** du menu contextuel du bouton droit de la souris.

Si vous sélectionnez une ou des cellules et choisissez **Supprimer les cellules...** du menu contextuel du bouton droit de la souris, vous pouvez choisir si les autres cellules de la feuille seront déplacées vers le haut ou vers la gauche pour emplir l'espace laissé par les cellules que vous avez choisi d'enlever.

Si vous voulez insérer une nouvelle ligne ou colonne vide dans la feuille, sélectionnez les lignes ou colonnes où vous voulez que les nouvelles lignes ou colonnes soient placées et choisissez l'option **Insérer des ligne(s)...**, **Insérer des colonnes...** du menu contextuel du bouton droit de la souris.

Vous pouvez insérer de nouvelles cellules dans la feuille en sélectionnant la zone où vous voulez qu'elles apparaissent, puis en choisissant l'option **Insérer des cellules...** du menu contextuel du bouton droit de la souris. On vous demandera si les cellules existantes de la zone sélectionnée doivent être déplacées vers le bas ou la droite pour faire de la place aux nouvelles.

## 2.6 Sommes simples

Si le premier caractère dans une cellule est un signe égal (=), Calligra Sheets considérera le contenu de la cellule comme une formule qui doit être calculée. Le résultat du calcul sera affiché dans la cellule plutôt que la formule elle-même. Par exemple, saisissez **=2+3** dans une cellule et elle devrait afficher 5.

Plus utilement, une formule peut contenir des références aux autres cellules, ainsi **=B4+A3** calculera la somme des valeurs des cellules B4 et A3 et ce calcul sera mis à jour lorsque les cellules B4 ou A3 seront modifiées.

De la même manière que pour l'addition, une formule peut faire usage du symbole - pour la soustraction, \* pour la multiplication et / pour la division. Les symboles de parenthèses arrondies ( ) peuvent aussi être utilisés comme dans l'algèbre normal, ainsi vous pouvez saisir des formules plus complexes comme **=(B10 + C3) \*5 - F11) / 2** .

Les cellules qui contiennent une formule seront marquées par un petit triangle bleu dans le coin en bas à gauche si la case **Afficher l'indicateur de formules** de la page **Format** → **Feuille** → **Propriétés de feuille** est coché.

Calligra Sheets inclut aussi un grand nombre de fonctions intégrées pour les applications comme les statistiques, la trigonométrie et les calculs financiers. Leur usage sera examiné plus en profondeur dans une [section ultérieure](#) de ce manuel, mais si vous êtes intéressé à ce stade, choisissez

**Fonction...** dans le menu **Insertion** et jetez un coup d'œil à la boîte de dialogue **Fonction** qui sera affichée.

Pour le moment, cependant, la fonction **SUM** peut être intéressante car elle calcule la somme de toutes les valeurs dans une zone spécifiée de cellules. Par exemple **=SUM(B4:C10)** calcule la somme de toutes les valeurs dans la zone de cellules allant de B4 à C10.

Si Calligra Sheets affiche **#VALUE!** quand vous avez saisi une formule, ceci signifie habituellement qu'il ne peut pas comprendre ce que vous avez saisi, mais si la ligne de symboles se termine par une petite flèche rouge, ceci signifie simplement que la cellule n'est pas assez grande pour afficher le résultat complet. Dans ce cas, vous devez soit élargir la cellule, soit changer son **format** pour que le résultat y tienne correctement.

## 2.6.1 Re-calcul

Si la case **Re-calcul automatique** de la page **Format** → **Feuille** → **Propriétés de feuille** est cochée, Calligra Sheets recalculera les valeurs des cellules au cas où quelque chose qui les affecte aurait changé.

Quand la case **Re-calcul automatique** n'est pas cochée pour la feuille courante, vous pouvez dire à Calligra Sheets d'effectuer un recalcul n'importe quand en utilisant les options **Recalculer la feuille** ou **Recalculer le document de travail** du menu **Outils** ou leurs raccourcis **Maj+F9** ou **F9**.

## 2.7 Tri des données

Dans le simple exemple ci-dessous, les données consistent en les noms et pays d'un certain nombre de montagnes avec leur hauteur par rapport à la mer. Calligra Sheets peut trier ces données comme ceci de différentes manières.

	A	B	C	D
1	<b>Montagnes</b>	<b>Pays</b>	<b>Hauteur (m)</b>	
2	McKinley	Alaska	6,194	
3	Roosevelt	Canada	2,972	
4	Kilimanjaro	Tanzanie	5,895	
5	Waddington	Canada	4,042	
6	Everest	Népal	8,848	
7	Roberts	Australie	1,387	
8				
9				
10				

Vous pouvez souhaiter trier les noms par ordre alphabétique. Pour le faire, sélectionner l'ère contenant les données (A2 :C7 dans ce cas) et choisir dans le menu **Trier...** depuis **Données**. Ceci ouvrira la boîte de dialogue **Trier**.

Le tri est fait d'une manière alphanumérique et est sensible à la casse, les nombres venant avant les majuscules, qui viennent avant les lettres minuscules, de telle manière que les cellules qui contiennent les entrées **Cat**, **bar**, **77** et **Bat** seront triées dans l'ordre suivant : *77 Bat Cat bar*.

Dans la partie nommée **Direction** de cette boîte de dialogue, choisissez de trier les lignes ou les colonnes. Si vous cochez **La première ligne contient les en-têtes** ou **La première colonne contient les en-têtes**, les données de la première ligne ou de la première colonne ne seront pas incluses dans l'opération de tri.

Les lignes et les colonnes sont triées dans l'ordre spécifié, qui peut être modifié en utilisant les boutons **Déplacer vers le haut** et **Vers le bas**. En utilisant l'exemple dans la capture d'écran ci-dessous, en choisissant la colonne B comme première clé et la colonne C comme seconde, les données seront triées par pays, puis pour chaque pays, par hauteur.

Décochez l'option **Sensible à la casse** pour obtenir un tri ne dépendant pas de la casse (majuscules / minuscules), et basculez entre les ordres de tri **Ascendant** et **Descendant** en cliquant sur la cellule dans la colonne **Ordre de tri**.

L'extension **Détail >>** de la boîte de dialogue vous permet de trier en utilisant les éléments d'une liste personnalisée comme Janvier, Février... au lieu d'une liste alphanumérique. Le format de cellule est déplacé avec le contenu de la cellule, si vous sélectionnez **Copier format de cellule** (**Bordures, Couleurs, Style de texte**).

## 2.8 La calcullette de sommaire de la barre d'état

L'extrémité gauche de la barre d'état affiche par défaut un sommaire des valeurs dans les cellules sélectionnées. Selon le réglage de la liste combinée **Fonctions montrées dans la barre d'état** dans la page **Interface** de la boîte de dialogue de configuration de Calligra Sheets, ce sommaire peut être :

### **Aucun**

Aucun calcul n'est effectué

### **Moyenne**

La valeur affichée est la moyenne des valeurs dans les cellules sélectionnées.

### **Compte**

La valeur affichée est le nombre de cellules contenant des valeurs numériques.

### **CountA**

La valeur affichée est le nombre de cellules non vides.

### **Max**

La valeur affichée est le maximum des valeurs dans les cellules sélectionnées.

### **Min**

La valeur affichée est le minimum des valeurs dans les cellules sélectionnées.

### **Somme**

La valeur affichée est la somme des valeurs dans les cellules sélectionnées.

La méthode de calcul peut aussi être changée en cliquant du bouton droit sur le libellé de résultat de calcul de la barre d'état et en choisissant un élément dans le menu contextuel.

## 2.9 Enregistrer votre travail

Calligra Sheets enregistre le document complet, qui peut inclure plusieurs feuilles, comme un simple document.

Si vous avez créé un document ou voulez en enregistrer un existant sous un nom différent, utilisez **Fichier** → **Enregistrer sous...** Ceci ouvrira la boîte de dialogue commune de KDE **Enregistrer le document sous**. Choisissez le dossier où vous voulez enregistrer le document et donnez un nom convenable dans la zone de texte **Nom** :. Les documents Calligra Sheets sont normalement



automatiquement enregistrés avec une extension `.ksp`, vous n'avez pas besoin d'ajouter ceci au nom du fichier, mais assurez-vous que la sélection **Filtre** : est positionnée sur **feuille de calcul ODS**.

Pour enregistrer votre document sans changer son nom, utiliser simplement **Fichier** → **Enregistrer**.

Vous pouvez également enregistrer un document Calligra Sheets dans un format étranger en choisissant le format dans la liste déroulante **Filtre** :

Quand vous avez enregistré une version modifiée d'un document existant, Calligra Sheets conservera la précédente version comme fichier de sauvegarde, ajoutant un `~` à la fin du nom du fichier.

Calligra Sheets peut fournir une certaine protection contre la perte de votre travail du fait d'un arrêt anormal de l'ordinateur ou de la fermeture de Calligra Sheets sans enregistrer le document de travail courant. Il fait cela automatiquement en enregistrant la dernière version du document sur lequel vous travaillez toutes les quelques minutes en utilisant un nom de fichier modifié. La version enregistrée automatiquement est normalement supprimée quand vous enregistrez ensuite votre document, de telle manière qu'il n'existe que s'il est plus à jour que la version enregistrée à la main. Quand vous ouvrez un document, Calligra Sheets vérifie s'il existe une version enregistrée automatiquement, et s'il en trouve une, il offrira de l'ouvrir à la place.

Les documents enregistrés automatiquement sont enregistrés avec un nom de fichier de la forme `.votre-nom-de-fichier.autosave` (notez le point au début), ainsi `spread1.ksp` serait enregistré automatiquement sous le nom `.tableur1.ksp.autosave`. Cette fonctionnalité d'enregistrement automatique est [configurable par l'utilisateur](#).

## 2.9.1 Modèles

Si vous allez créer beaucoup de documents semblables, vous pouvez économiser du temps et des soucis en créant d'abord un modèle, puis en l'utilisant comme base pour les documents individuels.

Pour faire ceci, créez d'abord un document contenant les éléments communs, puis enregistrez-le comme modèle en choisissant **Fichier** → **Créer un modèle à partir du document**. En faisant cela, s'ouvre la boîte de dialogue **Créer un modèle**. Saisissez un nom pour votre nouveau modèle dans la zone de texte **Nom** : et actionnez le bouton **Ok**. La prochaine fois que vous commencerez un nouveau document et choisissant **Fichier** → **Nouveau** ou quand vous démarrerez Calligra Sheets, la boîte de dialogue **Choisir** vous donnera l'option de créer le nouveau document à partir de votre modèle.

La boîte de dialogue **Créer un modèle** vous permet aussi de choisir une image différente à afficher au-dessus du nom du modèle dans la boîte de dialogue **Choisir** et vous permet d'enregistrer vos modèles sous différents groupes de noms, qui apparaîtront comme différentes pages à onglets dans la boîte de dialogue **Choisir**.

Les modèles sont stockés dans des fichiers `.kst` dans `~/ .kde/share/apps/tables/templates/` .

## 2.10 Imprimer un tableau

Imprimer un tableur se fait schématiquement en sélectionnant **Fichier** → **Imprimer...** qui amène la boîte de dialogue **Imprimer** de KDE, où vous pouvez choisir entre autres options, l'imprimante à utiliser, le nombre de copies et si toutes ou seulement les pages sélectionnées seront imprimées.

Par défaut, Calligra Sheets imprimera tous les éléments dans la feuille courante, mais vous pouvez restreindre ceci en sélectionnant d'abord la zone que vous voulez imprimer, puis en choisissant **Définir la plage d'impression** du sous-menu **Format** → **Plage imprimable**.

Calligra Sheets imprimera autant de feuilles que nécessaire pour inclure tous les éléments de la page courante. Vous pouvez voir rapidement comment un tableau sera coupé en pages séparées

pour l'impression en cochant la case **Affichage** → **Limites de pages**. Les limites de chaque page imprimée seront marquées par des lignes de couleur dans la feuille.

Pour une vue plus détaillée de ce qui sera envoyé à l'imprimante, y compris tout ce que vous avez demandé à être inclus dans les en-têtes et pieds de page (voir ci-dessous), choisissez **Fichier** → **Aperçu avant impression...**

Pour améliorer l'apparence de la sortie imprimée, vous pouvez changer les polices, les couleurs, les bordures et la taille des cellules dans la feuille. Voyez la section [Formatage du tableur](#) pour plus de détails sur le sujet.

Vous pouvez aussi utiliser la boîte de dialogue **Mise en page** invoquée en sélectionnant **Format** → **Format du papier...** pour changer le sens des pages imprimées, la taille du papier (cela devrait convenir à votre imprimante) et la taille des marges.

La **feuille** fournit davantage d'options. La section **Configuration d'impression** vous permet d'indiquer s'il faut ou non imprimer le quadrillage, les indicateurs de commentaires et de formules, les objets et les graphiques. La section **Répéter sur chaque page** vous permet de répéter les colonnes et les lignes sélectionnées sur chaque page imprimée. Dans la section **Dimensionnement**, vous pouvez définir un facteur d'échelle ou les limites de page pour l'impression.

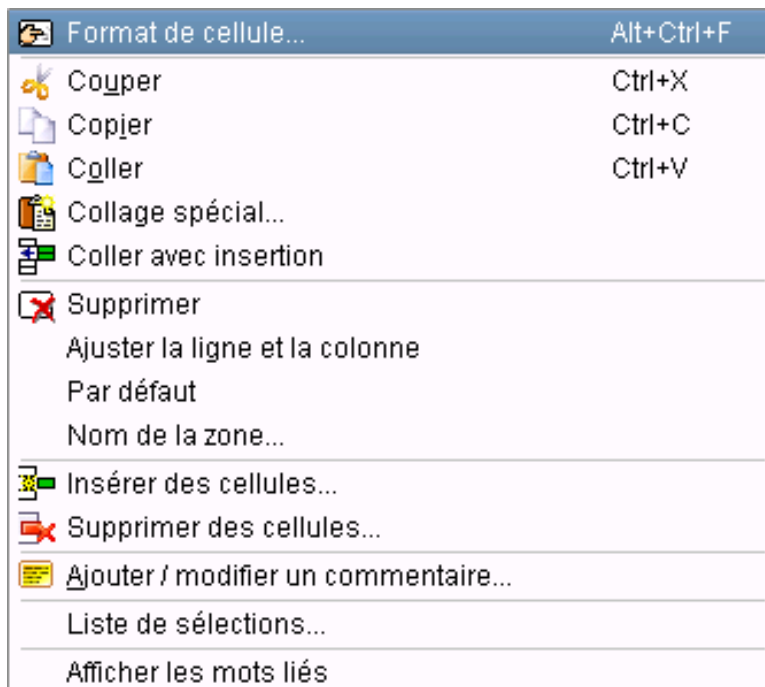
## Chapitre 3

# Formatage du Tableur

Pamela Robert  
Raphael Langerhorst  
Anne-Marie Mahfouf  
Traduction française : Gérard Delafond  
Traduction française : Daniel Durand  
Traduction française : Ludovic Grossard

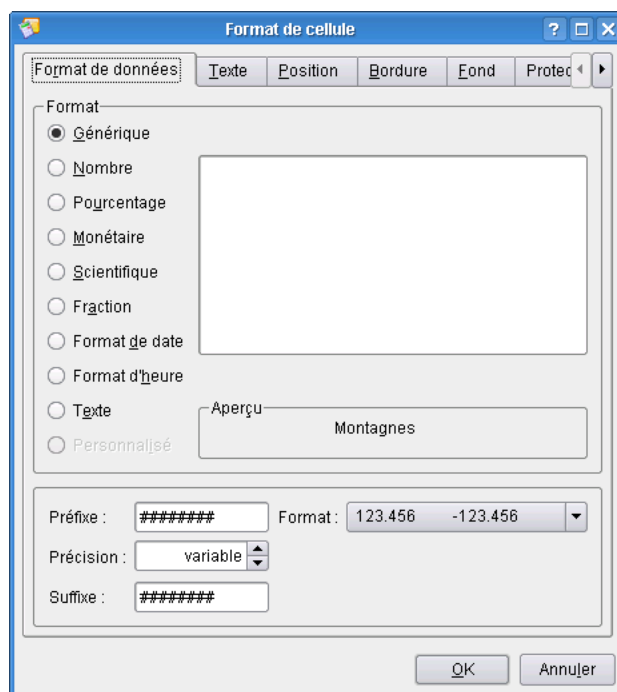
### 3.1 Format des cellules

Pour changer le format et l'apparence des cellules, lignes ou colonnes sélectionnées, utilisez l'option **Format de cellule...** du menu **Format** ou du menu contextuel du bouton droit de la souris.



Ceci ouvrira la boîte de dialogue **Format de cellule** qui présente plusieurs pages à onglets :

### 3.1.1 Formats de données et représentation




La page **Format de données** de la boîte de dialogue de **format de cellule** vous permet de contrôler comment les valeurs des cellules sont affichées.


La partie du haut de cette page vous permet de sélectionner le format à utiliser en affichant les valeurs numériques, les dates ou les heures. Un panneau **d'aperçu** vous permet de voir l'effet du nouveau format.

Vous pouvez définir le même format de date pour une ligne ou une colonne en sélectionnant la ligne ou la colonne et en ouvrant la boîte de dialogue **Format de cellule** avec le bouton droit de la souris.

**NOTE**  
 Vous pouvez augmenter la précision décimale de n'importe quel nombre pour les formats **Générique**, **Nombre**, **Pourcentage**, **Monétaire** or **Scientifique** en utilisant l'icône **Augmenter la précision** dans la barre d'outils **Format** :



Vous pouvez diminuer la précision décimale de n'importe quel nombre pour les formats **Générique**, **Nombre**, **Pourcentage**, **Monétaire** or **Scientifique** en utilisant l'icône **diminuer la précision** dans la barre d'outils **Format** :



#### Générique

Ceci est le format par défaut et Calligra Sheets détecte automatiquement le type de données suivant le contenu de la cellule. Par défaut, Calligra Sheets justifie à droite dans une cellule les nombres, les dates et les heures et justifie à gauche le reste.

Si le format **Générique** ne vous convient pas, vous pouvez le modifier pour un format particulier parmi les choix ci-dessous.

#### Nombre

La notation des nombres utilise la notation que vous avez choisie dans la configuration du système dans **Localisation** → **Pays / Région & Langue** → **Nombres**. Les nombres sont justifiés à droite par défaut.

### Pourcentage

Quand il y a un nombre dans la cellule courante et que vous souhaitez modifier le format de cellule de **Générique** à **Pourcentage**, le nombre dans la cellule est multiplié par 100.

Par exemple, si vous saisissez 2 et fixez le format de cellule à **Pourcentage**, le nombre deviendra 200 %. Modifier à nouveau le format de cellule à **Générique** le ramènera à 2.



Vous pouvez également utiliser l'icône **Pourcentage** dans la barre d'outils **Format** :

### Monétaire

Le format **Monétaire** convertit votre nombre en une notation correspondante à celle que vous avez choisie dans la configuration du système dans **Localisation** → **Pays / Région & Langue** → **Monétaire**. Le symbole de la devise est affiché et la précision peut être paramétrée dans la configuration du système.

Vous pouvez aussi utiliser l'icône **Format monétaire** dans la barre d'outils **Format** pour configurer la cellule conformément à votre devise actuelle :

### Scientifique

Le format **Scientifique** convertit votre nombre en une notation scientifique. Par exemple, 0.0012 sera converti en 1.2E-03. Si vous revenez au format de cellule **Générique** 0.0012 sera affiché de nouveau. Le format de cellule **Générique** ne conserve pas la notation scientifique, aussi si vous souhaitez conserver cette notation scientifique, vous devez le spécifier en utilisant le menu.

### Fraction

Le format **Fraction** convertit votre nombre en une fraction. Par exemple, 0.1 peut être modifié en 1/8, 2/16, 1/10, etc. Vous définissez le type de fraction en le choisissant dans le champ sur la droite. Si la fraction exacte n'est pas disponible dans le mode choisi, la concordance la plus proche est choisie. Par exemple, quand nous avons 1.5 comme nombre, nous choisissons **Fraction** et **1/16 un seizième**. Le texte affiché dans la cellule est « 1 8/16 » qui est la fraction exacte. Si vous avez 1/4 comme nombre dans votre cellule et si vous choisissez **Fraction** et **1/16 un seizième**, la cellule affichera « 1 6/16 » qui est la fraction en seizième la plus proche.

### Date

Pour insérer une date, vous devez le faire dans un des formats défini dans la configuration du système dans **Localisation** → **Pays / Région & Langue** → **Date et heure**. Il y a deux formats définis ici : le format long de date et le format court de date.

Un nombre naturel aléatoire « NN » sera transformé en date à partir du 30 décembre 1899 (qui est 0) avec le nombre de jours NN ajouté. Par exemple, si vous avez une cellule avec 100 et que vous choisissiez **Format de date**, « 1900-04-09 » s'affichera, ce qui correspond à 100 jours après le 30 décembre 1899. Cette date de départ est en avance de deux jours sur le calcul, car il y a eu un bug dans Lotus 123, qui est resté ainsi dans Excel pour des raisons de compatibilité. Peu de gens, de toute manière, ont besoin de calculer depuis le 1er janvier 1900 et ajouter 9 jours au 1er novembre 2000 par exemple, vous indiquera le 10 novembre 2000. En fait, tous les calculs de date sont corrects.

#### NOTE

Quand une cellule est au **Format de date**, vous pouvez étendre cette cellule vers le bas comme vous le faites avec les nombres et les cellules proches contiennent également des dates, chaque date étant incrémentée d'un jour.

### Heure

Cette option formate le temps de votre contenu de cellule. Pour saisir une heure, vous devez le faire dans le **format d'heure** défini dans la configuration du système **Localisation** → **Pays / Région & Langue** → **Date et Heure**. Dans la boîte de dialogue **Format de cellule** vous pouvez définir la manière d'afficher la date et l'heure en choisissant une des options. Le format par défaut est celui du système choisi dans la configuration du système. Quand le nombre dans la cellule n'a pas de sens en temps que date, Calligra Sheets affiche 00 :00 dans le format général existant dans la configuration du système.

### Date et heure

Cette option formate la date et l'heure de votre contenu de cellule. Pour saisir une date et une heure, vous devez ouvrir les paramètres **Date et heure** dans la configuration du système dans **Localisation** → **Pays / Région & Langue** → **Date et Heure**. Dans la boîte de dialogue **Format de cellule** vous pouvez définir la manière d'afficher la date et l'heure en choisissant une des options. Le format par défaut est celui du système choisi dans la configuration du système. Quand le nombre dans la cellule n'a pas de sens en temps que date, Calligra Sheets affiche 00 :00 dans le format général existant dans la configuration du système.

### Texte

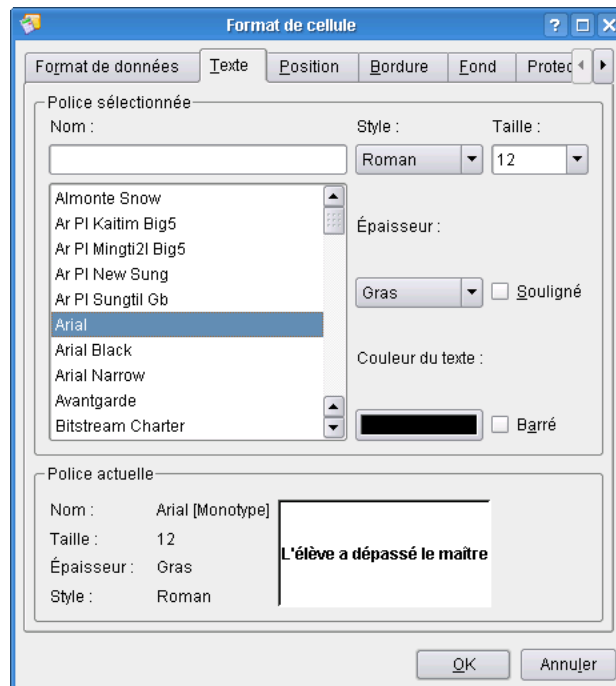
Cela formate le contenu de votre cellule en texte. Cela peut être utile si vous souhaitez qu'un nombre soit traité comme du texte et non comme un nombre. Par exemple, pour un code ZIP, configurer un nombre au format texte se justifie. Quand les nombres sont convertis au format texte, ils ne peuvent pas être utilisés dans les calculs ou les formules. Cela change également la manière dont la cellule est justifiée.

### Personnalisé

Ne fonctionne pas encore. Sera activé dans la prochaine mise à jour.

Le bas de la page **Données**, vous permet d'ajouter un **Préfixe** comme le symbole \$ au début de chaque élément ou un **Suffixe** comme \$HK à la fin. Vous pouvez aussi contrôler combien de chiffres sont affichés après le séparateur décimal pour les valeurs numériques, si les valeurs positives sont affichées précédées d'un signe + et si les valeurs négatives sont affichées en rouge.

## 3.1.2 Réglages des polices et du texte



L'onglet **Police** vous permet de sélectionner le style, la taille et la couleur de la police, y compris des options supplémentaires, comme le soulignement ou le texte barré. La partie du bas donne un aperçu du format de texte sélectionné.

La police par défaut est définie pour toutes les cellules dans le menu **Format** → **Gestionnaire de style** ainsi que le style courant.

**Style :**

Choisir le style de police pour les cellules sélectionnées. Quand vous sélectionnez plusieurs cellules de différents styles, le style affiché est défini dans **Variations (Sans changement)** et accepter cette option vous permet de conserver vos paramètres pour chaque cellule. Modifier en **Roman** par exemple modifiera tous les styles des cellules sélectionnées en **Roman**.

**Taille :**

Choisir la taille de la police de la cellule sélectionnée. Quand vous sélectionnez plusieurs cellules avec différentes tailles de polices, la taille affichée est définie dans (pas de chiffre affiché) et la laisser garde les paramètres actuels pour chaque cellule. Modifier en **14** par exemple modifie la taille de la police de toutes les cellules sélectionnées en **14**.

**Graisse :**

Choisir l'épaisseur de la police de la cellule sélectionnée. Quand vous sélectionnez plusieurs cellules avec différentes épaisseurs de polices, l'épaisseur affichée est définie dans **Variable (aucune modification)** et la laisser garde les paramètres actuels pour chaque cellule. Modifier en **Gras** par exemple modifie l'épaisseur des polices de toutes les cellules sélectionnées en **Gras**.

**Couleur :**

Choisir la couleur pour le texte des cellules actuellement sélectionnées. Cliquer sur la barre des couleurs ouvrira la boîte de dialogue standard de KDE **Sélectionner couleur** dans laquelle vous pourrez choisir la nouvelle couleur.

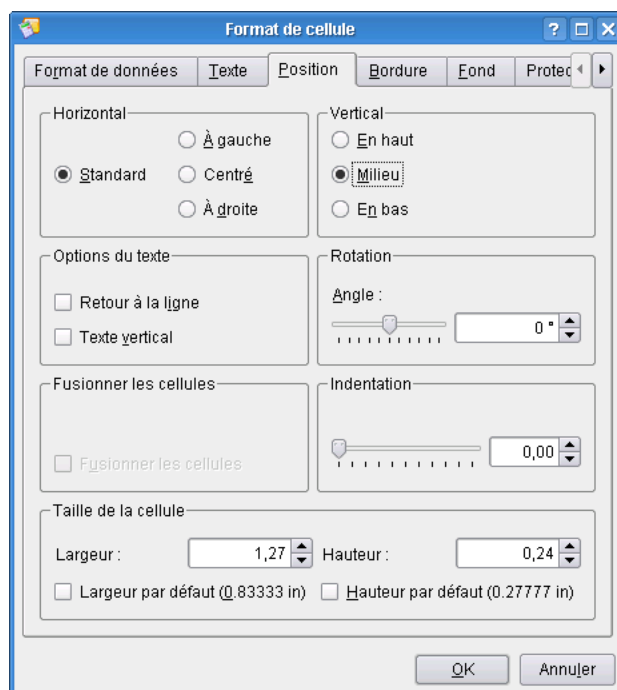
**Souligné :**

Souligne le texte des cellules sélectionnées si l'option est cochée. Elle n'est pas cochée par défaut.

**Barrer**

Cela barrera le texte des cellules sélectionnées si cette option est cochée. Elle n'est pas cochée par défaut.

### 3.1.3 Position et rotation du texte



Sur la page **Position**, vous pouvez contrôler la position du texte à l'intérieur d'une cellule en faisant les sélections convenables dans les zones **Horizontal** et **Vertical** ou en réglant la valeur

d'**Indentation**. Vous pouvez aussi choisir de faire que le texte apparaisse verticalement plutôt qu'horizontalement ou même avec un angle.

#### **Horizontal**

Définir horizontalement le contenu de la cellule. **Standard** est le choix par défaut et est déterminé par le format de données choisi. **Gauche** signifie que le contenu sera affiché à gauche dans la cellule. **Centre** signifie que le contenu sera centré dans la cellule. **Droite** signifie que le contenu sera affiché à droite dans la cellule.

#### **Vertical**

Définir verticalement le contenu de la cellule. **Haut** signifie que le contenu sera affiché en haut de la cellule. **Milieu** signifie que le contenu sera affiché au milieu de la cellule. **Bas** signifie que le contenu sera affiché en bas de la cellule.

#### **Options du texte**

Ceci est valable seulement quand la rotation est de 0. **Retour à la ligne** fait revenir le texte à la ligne afin qu'il tienne dans la taille de la précédente cellule. S'il n'est pas coché, le texte restera sur une seule ligne.

**Texte vertical** place votre texte verticalement

#### **Rotation**

Votre texte apparaîtra orienté selon l'angle que vous avez défini. Des valeurs positives le déplaceront dans le sens contraire aux aiguilles d'une montre et des valeurs négatives dans le sens des aiguilles d'une montre.

#### **Fusionner les cellules**

Quand cette option est cochée, elle fournit le même résultat que **Format** → **Fusionner les cellules**. Vous devez sélectionner au moins deux cellules consécutives. Ces cellules consécutives sont ensuite fusionnées dans une seule plus grande.

Quand une cellule fusionnée est sélectionnée et quand vous décochez cette option, toutes les cellules reviennent à leur taille originale d'avant la fusion. C'est le même effet que **Format** → **Dissocier les cellules**.

#### **Indenter**

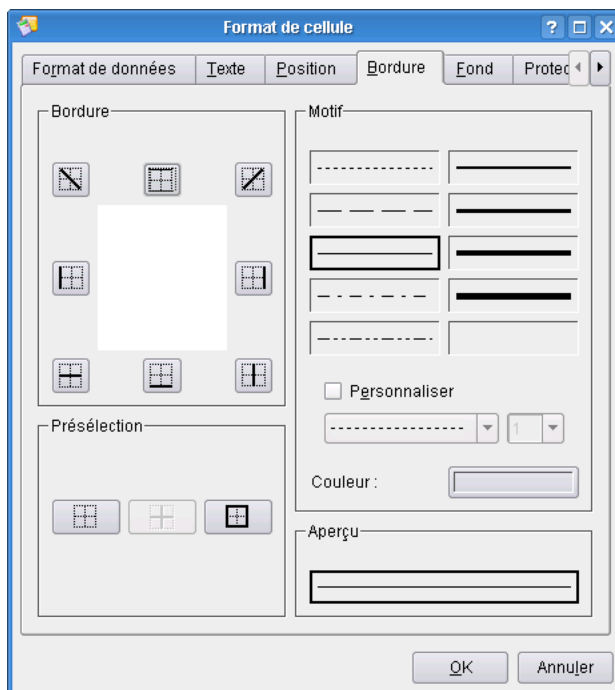
Règle la taille de l'indentation qui sera utilisée dans une cellule lorsque vous choisirez les actions **Augmenter l'indentation** et **Diminuer l'indentation** depuis la barre d'outils. Ces actions ne sont pas activées par défaut dans la barre d'outils.

#### **Taille de cellule**

Vous définissez ici la taille de la cellule, soit une largeur et une hauteur personnalisées, soit une largeur et une hauteur par défaut.



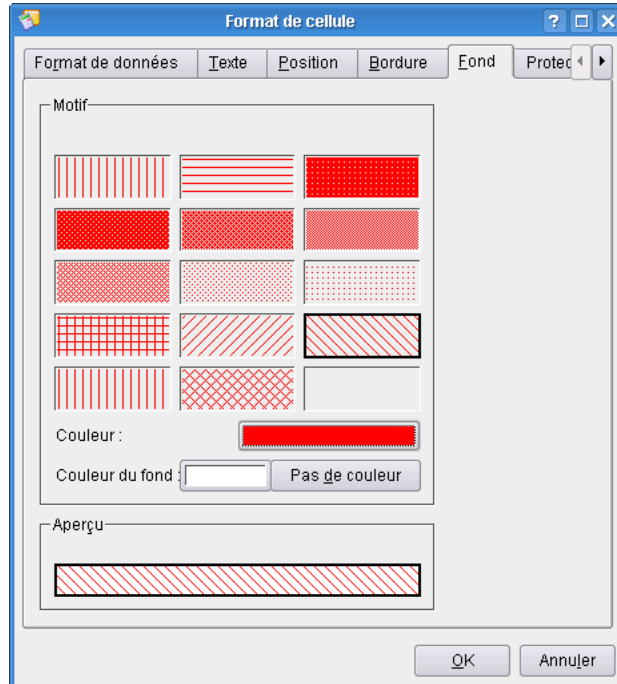
### 3.1.4 Bordure de cellule



La page **Bordure** vous permet de fixer l'apparence des bords de cellule. Si vous avez sélectionné plus d'une cellule, vous pouvez appliquer différents styles aux bordures entre les cellules et l'environnement de la zone sélectionnée.

Sélectionnez d'abord le motif et la couleur dans la section **Motif** de la page **Bordures**, puis appliquez-les aux différentes parties de la bordure en cliquant sur le bouton approprié de la section **Bordures**, ou sur l'un des boutons **Présélection**. Le bouton à gauche dans la section **Présélection** efface toute bordure précédemment appliquée. Notez que vous pouvez aussi ajouter une barre diagonale aux cellules.

### 3.1.5 Arrière-plan de cellule

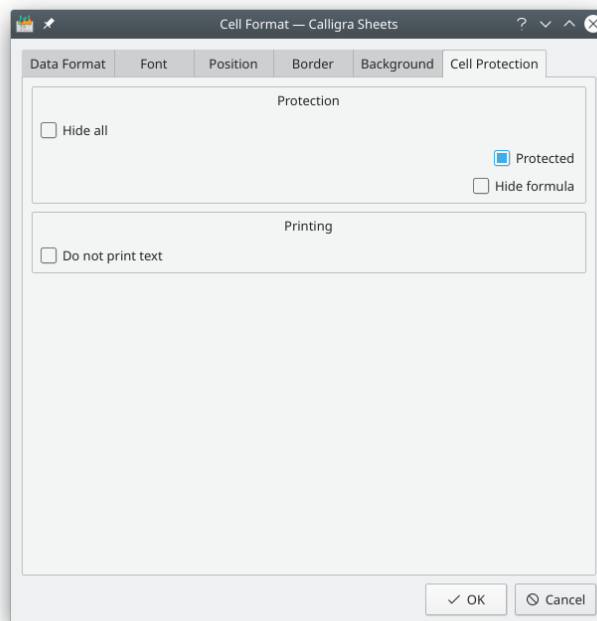


Le motif et la couleur du fond peuvent être sélectionnés dans la page **Fond**.

Choisissez simplement le **Motif** désiré, puis sélectionnez la couleur du motif et la couleur du fond.

En bas de cette page, vous pouvez voir un **Aperçu** du fond de la cellule configurée.

### 3.1.6 Protection de cellule



Vous pouvez modifier la manière dont le contenu d'une cellule est protégé dans la page **Protection de cellule**.

Toutes les cellules sont protégées par défaut (cela signifie que le contenu d'une cellule ne peut être modifié) et pour que la protection d'une cellule soit active, vous devez également protéger la feuille en utilisant le menu **Outils** → **Protection de feuille...** et fournir un mot de passe. Vous pouvez également masquer les formules de la cellule dans le but de protéger votre méthode de calcul de la formule. Cela nécessite pour fonctionner d'activer la protection de la feuille. Vous pouvez masquer le contenu de la cellule avec **Masquer tout** ce qui nécessite aussi de protéger la feuille. Vous pouvez en savoir plus sur ces paramètres dans le chapitre [Calligra Sheets avancé](#), [Section protection](#).

#### Cacher tout

Cette option masque le contenu de la cellule et fonctionne seulement quand la feuille est protégée ce qui signifie que changer l'attribut **Masquer tout** d'une cellule n'a aucun effet tant que la feuille est protégée. Que la cellule elle-même soit protégée ou non n'a pas d'importance.

Quand **Masquer tout** est sélectionné, **Protégé** et **Masquer formules** sont désactivés. Lorsque la feuille est protégée, **Masquer tout** masque le contenu de la cellule et les formules et cela masque et protège le contenu de la cellule.

#### Protégé

Si l'option est cochée, le contenu de la cellule sera protégé. C'est la situation par défaut. Vous devez protéger la feuille entière en utilisant le menu **Outils** → **Protéger la feuille....**

#### Cacher la formule

Quand cette option est cochée, la cellule est encore visible. De toute manière, son contenu n'apparaît pas dans la barre **Formule**. Masquer les formules fonctionne seulement pour les cellules qui contiennent des formules. Ainsi, l'utilisateur ne peut voir les formules. La feuille doit être protégée ainsi pour être fonctionnelle.

#### Ne pas imprimer le texte

Si vous cochez **Ne pas imprimer le texte**, le texte dans la cellule ne sera pas imprimé. Cette option est décochée par défaut ce qui signifie que le texte des cellules est toujours imprimé par défaut.

## 3.2 Attributs conditionnels des cellules

Vous pouvez faire changer l'apparence d'une cellule selon la valeur qu'elle contient, peut-être utile si vous utilisez Calligra Sheets pour conserver une trace de vos dépenses personnelles et voulez surligner tout élément plus grand que mille Euros.

Pour faire cela, sélectionnez les cellules, puis choisissez **Styles conditionnels...** dans le menu **Format**. Ceci ouvre la boîte de dialogue **Styles conditionnels** où vous pouvez faire changer le type de police et la couleur de la cellule quand la valeur rencontre une ou plusieurs conditions. Notez que les deuxième et troisième conditions ne s'appliquent que si les précédentes ne sont pas rencontrées.

Utilisez **Effacer** → **Styles conditionnels** du menu **Édition** pour supprimer les attributs conditionnels des cellules sélectionnées.

## 3.3 Changer la taille des cellules

La page **Position** dans le dialogue **Format de cellule** vous permet de modifier la taille des cellules sélectionnées. Notez que changer la hauteur d'une seule cellule changera la hauteur de toutes les cellules d'une ligne, de même que changer la largeur d'une cellule affectera toute la colonne.

Vous pouvez aussi sélectionner les lignes ou colonnes à changer, puis sélectionner **Redimensionner la ligne...** ou **Redimensionner la colonne...** du menu contextuel du bouton droit de la souris ou du menu **Format** → **Ligne** ou **Format** → **Colonne**.

Si vous déplacez le pointeur de la souris de telle manière que son extrémité est sur la ligne entre deux des numéros de lignes à gauche de la fenêtre de Calligra Sheets, le pointeur se changera pour afficher deux lignes parallèles, chacune avec une petite flèche qui en part. Quand le pointeur est dans cet état, vous pouvez tenir enfoncé le bouton gauche de la souris et tirer le bord entre les deux lignes, changeant la hauteur de la ligne supérieure. Une technique semblable peut être utilisée pour changer la largeur d'une colonne.

Pour établir la hauteur de la ligne ou la largeur de la colonne au minimum nécessaire pour afficher le contenu, sélectionnez la ligne ou la colonne entière et cliquez du bouton droit sur le libellé de la ligne ou de la colonne. Dans le menu qui apparaît, sélectionnez **Ajuster la ligne** ou **Ajuster la colonne**. La ligne ou la colonne se redimensionnera au minimum nécessaire. Vous pouvez aussi sélectionner une simple cellule ou une plage de cellules et cliquer sur **Ajuster les lignes et colonnes**, soit du menu du bouton droit, soit dans le menu **Format**.

Vous pouvez donner la même taille à plusieurs lignes ou colonnes adjacentes en sélectionnant **Format** → **Ligne** → **Égaliser les lignes** ou **Format** → **Colonne** → **Égaliser les colonnes**.

### 3.4 Fusionner les cellules

Il est souvent pratique d'avoir une cellule qui s'étale sur deux ou plusieurs colonnes ou sur plusieurs lignes. Ceci peut se faire en fusionnant deux ou plusieurs cellules en une. Sélectionnez les cellules à fusionner, puis choisissez **Format** → **Fusionner les cellules**.

Pour inverser ce processus, sélectionnez la cellule fusionnée et choisissez **Dissocier les cellules** dans le menu **Format**.

### 3.5 Cacher lignes et colonnes

Un tableur bien fini peut souvent se faire de manière à avoir l'air plus attirant en cachant les cellules qui contiennent des calculs intermédiaires, de telle manière que seules les données importantes et les résultats sont affichés.

Dans Calligra Sheets, vous pouvez cacher les lignes ou colonnes sélectionnées en utilisant les options **Cacher les lignes** et **Cacher les colonnes** du menu **Format** → **Lignes**, **Format** → **Colonne** ou du menu contextuel du bouton droit de la souris. Les lignes cachées ne sont pas affichées sur l'écran ni incluses dans la sortie imprimante.

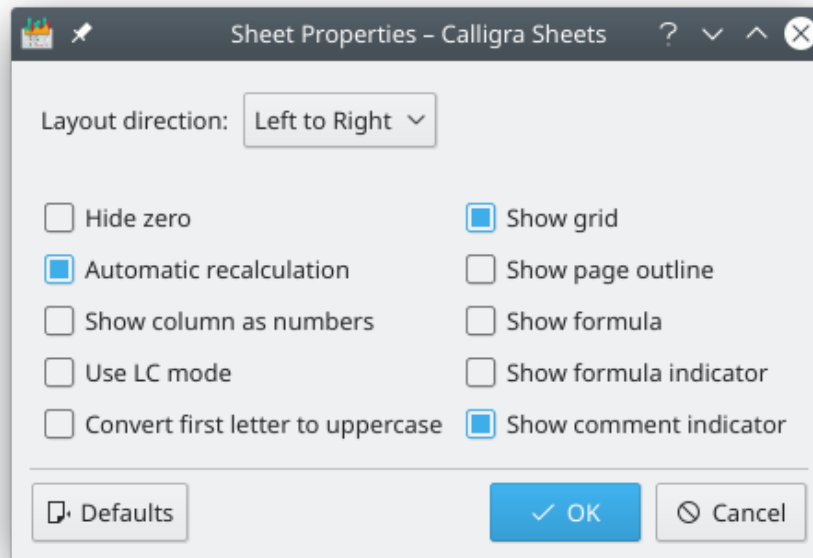
Cacher des cellules de cette manière les rend aussi moins vulnérables à une modification accidentelle.

Pour afficher à nouveau une ligne ou une colonne, sélectionnez **Ligne** → **Afficher les lignes...** ou **Colonne** → **Afficher les colonnes...** dans le menu **Format**. Dans la boîte de dialogue qui apparaît, vous pouvez sélectionner n'importe quel nombre de lignes à afficher.

### 3.6 Propriétés de la feuille

Vous pouvez accéder aux propriétés de la feuille courante, soit en cliquant-droit sur l'onglet feuille et en choisissant **Propriétés de la feuille** ou en utilisant le menu **Format** → **Feuille** → **Propriétés de la feuille**. Veuillez noter que vous pouvez seulement accéder à **Propriétés de la feuille** si le document ou la feuille ne sont pas protégés.

Vous pouvez définir différentes propriétés valables pour la feuille courante. Cliquer sur **Ok** rendra effectives vos modifications et **Défaut** restaurera les paramètres initiaux.



#### Mise en page :

Vous permet de choisir l'orientation de la feuille. Par défaut, la première colonne de la feuille est à gauche. Si vous choisissez **De droite à gauche**, alors la première colonne sera à droite et les autres seront ajoutées de la droite vers la gauche.

#### Cacher zéro

Si la boîte de dialogue est décochée, toute cellule contenant la valeur zéro apparaîtra blanche.

#### Re-calcul automatique

Ce paramètre vérifie que les formules sont recalculées automatiquement quand la valeur d'une cellule à laquelle elle se réfère est modifiée.

#### Afficher les colonnes au format nombre

Si cette case est cochée, l'en-tête de colonne affichera des nombres plutôt que des lettres. Les lettres sont affichées par défaut

#### Utiliser le mode LC

Si cette case est cochée, la référence de cellule affichée du côté gauche de la Barre de formules sera affichée en mode LC (i.e L2C3) plutôt que sous sa forme normale B3. Cela ne semble pas être d'une grande utilité jusqu'à présent.

#### Convertir la première lettre en majuscule.

Cocher cette case et la première lettre de tout texte saisi sera automatiquement convertie en capitale.

#### Afficher la grille

Si la case est cochée, la grille (les limites de cellule) sera affichée. C'est le choix par défaut. Si vous la décochez, la grille sera masquée.

#### Afficher les bordures de pages

Si vous cochez cette option, les bordures de pages seront tracées sur votre feuille courante. Par défaut, les bordures de pages ne sont pas affichées. Il peut être utile de voir les bordures de pages si vous souhaitez imprimer votre feuille.

#### Afficher les formules

Si cette case est cochée, Calligra Sheets affichera les formules actuelles dans les cellules plutôt que les résultats.

**Afficher l'indicateur de formule**

Si cette case est cochée, Calligra Sheets affichera un petit triangle bleu dans le coin inférieur gauche des cellules contenant des formules. Cela peut être utile si vous souhaitez protéger les cellules contenant des formules.

**Afficher l'indicateur de commentaires**

Si cette case est cochée, les cellules contenant des commentaires seront marquées avec un petit triangle rouge dans le coin supérieur droit.

## Chapitre 4

# Calligra Sheets expert

Pamela Robert  
Anne-Marie Mahfouf  
Traduction française : Daniel Durand  
Traduction française : Ludovic Grossard

### 4.1 Séries

En créant un tableur, vous aurez souvent besoin d'inclure une série de valeurs comme 10, 11, 12... en ligne ou en colonne. Il y a plusieurs manières de le faire dans Calligra Sheets.

Pour une simple série courte comme 5, 6, 7, 8... la méthode "Tirer et copier" est la plus simple. Saisissez la valeur de début dans la cellule de début et la valeur suivante dans la cellule adjacente. Puis, sélectionnez les deux cellules et déplacez le pointeur de la souris de telle manière qu'il soit sur le petit carré dans le coin en bas à droite de la cellule. La curseur se transformera alors en double flèche oblique. Maintenez le bouton gauche de la souris enfoncé pendant que vous tirez les cellules verticalement ou horizontalement comme nécessaire.

La taille du pas est calculée comme la différence entre les deux valeurs de départ que vous avez saisies. Par exemple si vous saisissez **4** dans la cellule A1 et **3,5** dans la cellule A2, sélectionnez les deux et tirez vers le bas. La taille du pas sera la valeur de A2 moins A1, -0,5 dans ce cas, et vous devriez donc obtenir la série 4 - 3,5 - 3 - 2,5 - 2...

La méthode "Tirer et copier" tiendra même compte d'une série où la valeur de pas n'est pas une valeur constante, mais lui-même une série. ainsi, si vous commencez par 1, 3, 4, 6, un tirer et copier l'étendra en 1, 3, 4, 6, 7, 9, 10, 12..., la valeur du pas dans cet exemple étant la série 2, 1, 2, 1...

Calligra Sheets reconnaît aussi certaines séries "spéciales" comme les jours de la semaine. Essayez d'écrire **Vendredi** dans une cellule (notez la majuscule) puis utilisez le Tirer et copier vers le bas. Pour voir quelles séries sont disponibles, et peut-être, créer les vôtres, sélectionnez **Outils** → **Listes personnalisées...**

Si vous sélectionnez une cellule et choisissez **Série...** du menu **Insertion**, vous verrez la boîte de dialogue **Séries**. C'est utile pour créer des séries qui sont trop longues pour être facilement construites en utilisant la méthode Tirer et Copier, ou pour créer des séries géométriques comme 1, 1.5, 2.25, 3.375... où la valeur du pas, 1,5 dans ce cas, est un multiplicateur.

Si le type de série que vous voulez est trop compliquée pour aucune des méthodes précédentes, utilisez une formule et faites un Tirer et Copier. Par exemple, pour créer une série avec les valeurs 2, 4, 16, 256... saisissez **2** dans A1, **=A1\*A1** dans A2, et faites un Tirer et Copier de la cellule A2 vers le bas.

## 4.2 Formules

### 4.2.1 Fonctions intégrées

Calligra Sheets a une énorme quantité de **fonctions** mathématiques et autres intégrées, qui peuvent être utilisées dans une cellule de formule. On peut les voir et les utiliser en sélectionnant une cellule et en choisissant **Fonction...** du menu **Insertion**. Ceci ouvre la boîte de dialogue **Fonction**.

Sélectionnez la fonction que vous voulez utiliser dans la boîte à liste à gauche de la boîte de dialogue, La page d'onglet **Aide** affiche la description, le résultat retourné, la syntaxe, les paramètres de cette fonction. En supplément, la page affiche souvent des liens vers des fonctions en relation. Ensuite actionner le bouton avec la flèche vers le bas pour coller l'expression dans la zone d'édition en bas de la boîte de dialogue.

La page à onglet **Paramètres** s'affichera alors pour vous permettre de saisir les paramètres pour la fonction que vous venez de choisir. Si vous voulez saisir une vraie valeur pour un paramètre, écrivez simplement dans la zone de texte appropriée dans la page **Paramètres**. Pour saisir une référence à une cellule plutôt qu'une valeur, cliquez du bouton gauche dans la boîte de texte appropriée dans la page **Paramètres**, puis cliquez du bouton gauche sur la cellule-cible dans la feuille.

Au lieu d'utiliser la page **Paramètres**, les références aux cellules comme **B6** peuvent être écrites directement dans la zone d'édition en bas de la boîte de dialogue **Fonction**. Si une expression a plusieurs paramètres, séparez-les par un point-virgule (;).

Cliquer le bouton **Ok** insérera la fonction dans la cellule courante et fermera la boîte de dialogue **Fonction**.

Vous pouvez, bien sur, faire sans la boîte de dialogue **Fonction** et simplement écrire l'expression complète dans la boîte de saisie de texte de l'**éditeur de cellule**. Les noms des fonctions ne sont pas sensibles à la casse. N'oubliez pas que toutes les expressions doivent commencer par un symbole =.

### 4.2.2 Comparaisons logiques

Les fonctions logiques comme IF(), AND(), OR() prennent des paramètres qui ont les valeurs logiques (booléennes) Vrai ou Faux. Ce type de valeur peut être produit par d'autres fonctions logiques comme ISEVEN() ou par comparaison de valeurs dans les cellules de tableur en utilisant les expressions de comparaison données dans le tableau suivant.

Expression	Description	Exemple
==	Est égal à	<b>A2==B3</b> est Vrai si la valeur dans A2 est égal à la valeur dans B3
!=	N'est pas égal à	<b>A2!=B3</b> est Vrai si la valeur dans A2 n'est pas égale à la valeur dans B3
<>	N'est pas égal à	Identique à <b>A2!=B3</b>
<	Est plus petit que	<b>A2&lt;B3</b> est Vrai si la valeur dans A2 est plus petite que la valeur dans B3
<=	Est inférieur ou égal à	<b>A2&lt;=B3</b> est Vrai si la valeur dans A2 est inférieure ou égale à la valeur dans B3
>	Est plus grand que	<b>A2&gt;B3</b> est Vrai si la valeur dans A2 est supérieure à la valeur dans B3



>=	Est plus grand ou égal à	<b>A2&gt;=B3</b> est Vrai si la valeur dans A2 est supérieure ou égale à la valeur dans B3
----	--------------------------	--

Ainsi, si vous saisissez **=IF (B3>B1; "Plus Grand"; "")** dans une cellule, il y aura affichage de Plus Grand si la valeur dans B3 est plus grande que B1, sinon, la cellule n'affichera rien.

### 4.2.3 Références absolues aux cellules

Si une formule contient une référence à une cellule, cette référence sera normalement changée quand la cellule est copiée vers un autre endroit de la feuille de travail. Pour empêcher ce comportement, ajoutez un symbole \$ avant la lettre de colonne ou avant le numéro de ligne.

- Si A1 contient la formule **=D5**, une copie sur la cellule B2 la transformera en **=E6** (le comportement normal).
- Si A1 contient la formule **=\$D5**, une copie sur la cellule B2 la transformera en **=D6** (la lettre de colonne inchangée).
- Si A1 contient la formule **=D\$5**, une copie sur la cellule B2 la transformera en **=E5** (le numéro de ligne inchangé).
- Si A1 contient la formule **=\$D\$5**, lors d'une copie sur la cellule B2, elle restera **=D5** (ni le numéro de ligne, ni la lettre de colonne n'est changé).

Quand vous saisissez ou éditez une référence de cellule dans une formule, la touche de raccourci **F4** peut être utilisée pour circuler parmi les quatre possibilités.

**Cellules nommées** peut être utilisé d'une manière semblable pour inclure une référence de cellule non changeante dans une formule.

## 4.3 Arithmétique en utilisant le collage spécial

Parfois, vous pouvez vouloir ajouter une valeur simple à un certain nombre de cellules ou en retirer une valeur, ou tout multiplier ou diviser par une valeur simple. L'option **Collage spécial...** vous permet de le faire vite et facilement.

Saisissez d'abord la valeur du modificateur dans n'importe quelle cellule de rechange sur votre tableur et la **Copier**. Puis, sélectionnez la zone de cellules que vous voulez modifier, choisissez **Collage spécial...** du menu **Édition** ou du menu contextuel du bouton droit de la souris et sélectionnez **Addition**, **Soustraction**, **Multiplication** ou **Division** de la section **Opérations** de la boîte de dialogue.

Vous pouvez aussi appliquer différentes valeurs de modificateurs à différentes lignes ou colonnes de la zone cible en copiant une zone contenant les modificateurs voulus avant de sélectionner la zone cible et de faire **Collage spécial...** Par exemple, si vous saisissez **5** dans la cellule A1, **10** dans B1, sélectionnez les deux cellules et faites un **Copier** puis **Collage spécial...Addition** dans les cellules A10 à D15, 5 seront ajoutés à A10 :A15 et C10 :C15, et 10 à B10 :B15 et D10 :D15.

Notez qu'une valeur de modificateur peut être une formule aussi bien qu'une valeur numérique simple. Si c'est une formule, Calligra Sheets ajustera les références des cellules comme pour une opération de **Coller** normale.

## 4.4 Formules de tableau

Calligra Sheets vous permet d'utiliser des formules dont le résultat est une matrice ou une plage de valeurs. Normalement, seule la première valeur est affichée dans la cellule. Si vous voulez afficher la totalité de la matrice, veuillez simplement utiliser **Ctrl-Alt-Entrée** lors de l'édition de la formule et elle sera convertie en une formule de tableau, occupant les cellules voisines lorsque nécessaire.

Les cellules faisant partie de la formule de tableau sont verrouillées pour modifications.

## 4.5 Recherche de solution

Calligra Sheets peut être utilisé pour résoudre des expressions algébriques comme  $x+x^2=4$  ou *Pour quelle valeur de x, x+x carré font 4 ?*.

Pour cet exemple, vous pourriez écrire **=A2+A2\*A2** dans A1, puis essayer différentes valeurs dans A2 jusqu'à ce que le résultat soit aussi près que vous le souhaitez de 4, ou, mieux, utiliser la fonctionnalité de Calligra Sheets **Recherche de solution** pour ajuster automatiquement la valeur dans une cellule pour essayer de la rendre la valeur dans une autre cellule aussi proche que possible de la valeur-cible.

Cette fonction est appelée en sélectionnant **Recherche de solution** dans le menu **Données**. Ceci ouvre une boîte de dialogue dans laquelle vous devez saisir la référence de la cellule qui doit avoir la valeur (**A1** dans ce cas), dans la zone **Définir la cellule**, la valeur-cible elle-même (**4** dans la zone **à la valeur** et la référence à la cellule à modifier (**A2**) dans la zone **Par changement de cellule**. Notez que vous avez besoin de saisir des valeurs initiales dans la cellule qui doit changer avant de commencer la **recherche de solution**.

En actionnant le bouton **Ok** dans le menu **Recherche de solution**, le calcul commencera. Quand il finira et s'il a trouvé une solution, actionnez le bouton **Ok** pour accepter le résultat ou **Annuler** pour conserver la valeur originale.

## 4.6 Tables avec pivot

Calligra Sheets peut être utilisé pour la création de **tables avec pivot** en utilisant les données de la table courante.

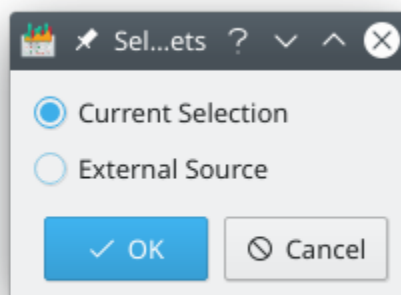
Cette tâche peut être démarrée facilement par sélection de **Pivot...** à partir du menu **Données**. Un exemple de génération de table avec pivot est donné ci-dessous.

En supposant que les données suivantes soient.

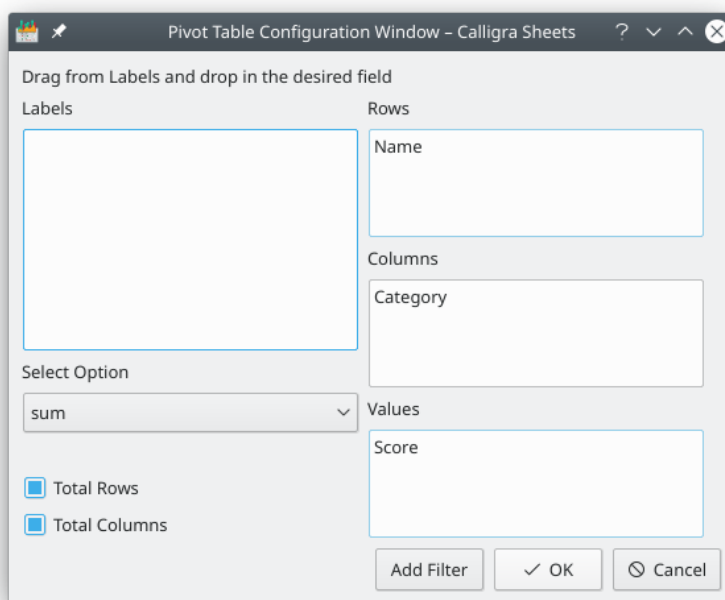
	A	B	C
1	Name	Category	Score
2	Jigar	Science	90
3	Smith	Math	80
4	John	Science	95
5	Smith	Science	60
6	Jigar	Math	81
7	John	Math	90

Pour la création d'une table avec pivot selon votre idée et besoins, sélectionner **Données** → **Pivot...**

La boîte de dialogue qui apparaîtra, permet à l'utilisateur de choisir la source de données. Les données peuvent être sélectionnées à partir de la feuille de travail courante ou à partir d'une source externe comme une base de données ou un fichier ODS.

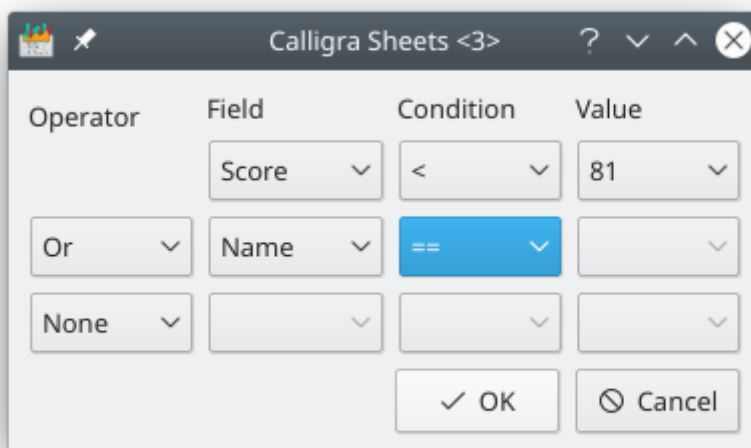


Cette boîte de dialogue permet à l'utilisateur de personnaliser la table avec pivot. Les étiquettes de colonne dans les données source sont converties en étiquettes, servant de champs. Les étiquettes peuvent être glissées et déposées dans l'une des trois zones (**Lignes**, **Colonnes** ou **Valeurs**) pour la génération de la table avec pivot. Vous pouvez ré-initialiser vos choix en utilisant le bouton **Ré-initialiser DnD**.



Dans l'exemple, *Nom* est glissé vers **Lignes**, *Catégorie* vers **Colonnes**, *Score* vers **Valeurs**. Les fonctions définies par l'utilisateur comme somme, moyenne, max, min, compteur, etc. peuvent être sélectionnées à partir de la liste **Sélectionner une option**.

Le bouton **Ajouter un filtre** peut être utilisé pour ouvrir une boîte de dialogue pour filtrer les données souhaitées. En utilisant cette dernière, vous pouvez définir de nombreux filtres reposant sur les étiquettes de colonnes ou les relations entre elles (**Et** ou **Ou**). Cela vous permettra la plus grande liberté pour personnaliser vos calculs.



**Nombre total de lignes** et **Nombre total de colonnes** : la vérification de ces variables vous permet automatiquement de traiter la totalité des lignes et des colonnes dans la table avec pivot.

## 4.7 Utiliser plusieurs feuilles de travail

Quand vous démarrez un nouveau document vide avec Calligra Sheets, il créera un certain nombre de feuilles vierges. Le nombre de feuilles qui sont créées est déterminé par le modèle choisi.

**Insertion** → **Feuille** ajoutera une nouvelle feuille au document.

Vous pouvez aussi basculer entre les feuilles en utilisant **Ctrl+Page suiv.** pour vous déplacer à la prochaine feuille, **Ctrl+Page préc.** pour vous déplacer à la précédente.

Il est donné aux feuilles les noms par défaut de *Feuille 1*, *Feuille 2*... Vous pouvez donner au tableau un nom différent en cliquant du bouton droit sur l'onglet et en sélectionnant **Renommer la feuille**...

Pour enlever une feuille du livret de travail, utilisez l'option **Enlever la feuille** dans le menu contextuel qui surgit quand vous cliquez avec le bouton droit sur l'onglet de la feuille que vous voulez enlever.

Les autres entrées du menu **Format** → **Feuille** vous permettent d'enlever une table ou de la masquer de la même manière que les lignes et les colonnes peuvent être masquées.

Si vous voulez qu'une formule dans une feuille se réfère à une cellule dans une autre feuille, la cellule de référence doit commencer par le nom de la feuille suivi d'un point d'exclamation (!). Par exemple, si vous saisissez **=Feuille2!A2** dans une cellule de la feuille 1, cette cellule prendra la valeur de A2 de la feuille 2. Notez que les noms de feuilles tiennent compte de la casse.

### 4.7.1 Consolider des données

Vous pouvez avoir construit un livret contenant plusieurs feuilles de travail, contenant des données semblables, mais, disons, différents par les mois de l'année, et souhaitez avoir une feuille de synthèse contenant les valeurs consolidées (par exemple la somme ou la moyenne) des valeurs des éléments correspondants dans les autres feuilles.

Cette tâche peut se faire facilement en utilisant l'élément **Consolider...** du menu **Données**.

La sélection de cette option ouvre la boîte de dialogue **Consolidation**.

Pour chacune des feuilles sources, saisissez une référence à la zone de données voulues dans la boîte **Références**. Actionnez **Ajouter** qui devrait la transférer dans la boîte des **Entrées référencées**. La référence devrait inclure le nom de la feuille contenant la source de données, comme **Janvier!A1:A10** et peut être saisie automatiquement en sélectionnant la zone dans le tableau approprié.

Quand vous avez saisi les références pour toutes les feuilles de données sources, sélectionnez la cellule dans la feuille-cible là où vous voulez que le résultat consolidé apparaisse, choisissez la fonction appropriée dans la liste déroulante **Fonction**, puis actionnez le bouton **Ok**.

Si vous cliquez sur le bouton **Détails >>** de la boîte de dialogue, et cochez la case **Copier les données**, les valeurs résultant de la consolidation seront placées dans les cellules cibles plutôt que les formules qui servent à les calculer.

## 4.8 Insérer un diagramme

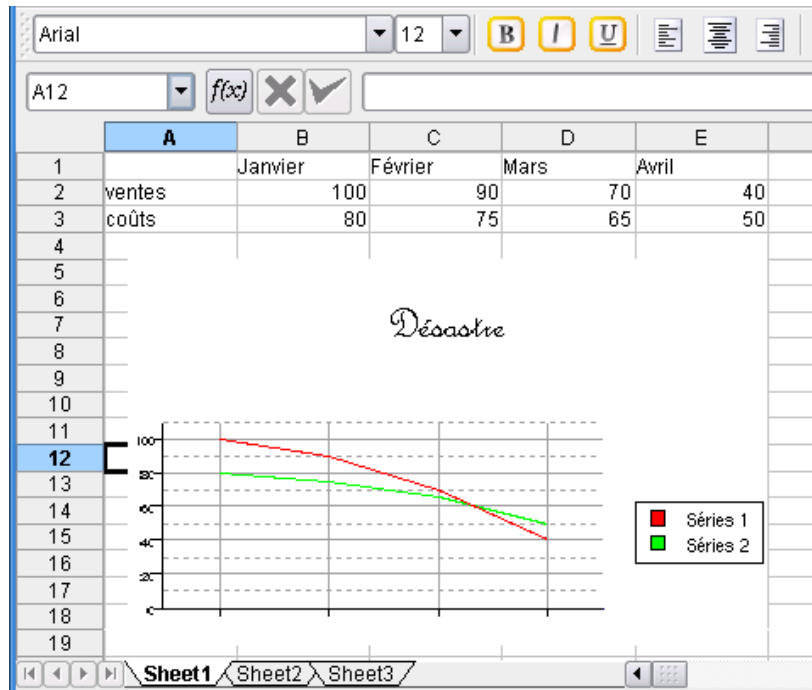
Vous pouvez insérer un diagramme dans une feuille pour donner une vue graphique à vos données.

Tout d'abord, activez le panneau **Ajouter une forme** en utilisant l'élément de menu **Paramètres** → **Panneaux**.

Sélectionnez ensuite la zone de cellules contenant les données et choisissez **Diagramme** dans le panneau **Ajouter une forme**. Glissez le curseur sur la feuille, tout en maintenant le bouton gauche de la souris enfoncé pour définir la zone où vous voulez que le graphique apparaisse; il n'y a pas besoin d'être précis pour le moment, car la taille du diagramme peut être facilement changée n'importe quand. Quand vous relâchez le bouton de la souris, la boîte de dialogue **Options diagramme** apparaît.

Les données sont préremplies avec la plage de cellules sélectionnées. Choisissez si vous voulez utiliser la première ligne et la première colonne comme en-tête, cochez **Configuration des données dans les lignes** et cliquez sur le bouton **Ok**. La boîte de dialogue disparaîtra et vous pourrez voir le diagramme intégré dans la feuille.

Sélectionnez maintenant **Outil d'édition des diagrammes** dans le panneau **Outils** et modifiez les propriétés du diagramme comme le type de tracé, les étiquettes et les axes dans **Modification du diagramme**.



Pour déplacer, redimensionner ou même supprimer le diagramme intégré, allez dans l'outil **Manipulations simples de formes** et cliquez n'importe où dans la zone de diagramme. Il devrait apparaître avec une bordure verte et avec de petits carrés jaunes à chaque coin et au milieu de chaque bord.

Si vous déplacez le curseur sur n'importe lequel des carrés, il devrait se transformer en une flèche à deux têtes. Vous pouvez redimensionner le diagramme en tirant l'un de ces carrés avec le bouton gauche de la souris enfoncé. Pour supprimer le diagramme, cliquez du bouton droit sur l'un des carrés et sélectionnez **Supprimer**.

Pour déplacer le diagramme, déplacez le curseur à l'intérieur de celui-ci. Le curseur se transformera en croix. Actionnez le bouton gauche de la souris et vous pourrez tirer le diagramme là où vous voulez qu'il aille.

Pour rendre au diagramme son apparence normale, cliquez simplement n'importe où en dehors de la zone de diagramme.

Pour modifier le format du diagramme lui-même, cliquez deux fois avec le bouton gauche dans la zone de diagramme. La **Modification du diagramme** devrait apparaître dans le panneau. Vous pouvez alors utiliser ces outils pour modifier le diagramme.

## 4.9 Insérer des données externes

Vous pouvez insérer des données d'un fichier texte ou depuis le presse-papiers dans une feuille en sélectionnant d'abord la cellule où vous voulez l'élément qui sera en haut à gauche, puis en choisissant **D'un fichier de texte...** ou **Du presse-papiers...** du sous-menu **Insertion** → **Données externes**.

Dans les deux cas, Calligra Sheets supposera que les données sont en format CSV et ouvrira une boîte de dialogue vous permettant de contrôler comment les données sont extraites du fichier ou du presse-papiers et placées dans les cellules de la feuille de travail.

Si votre système le prend en charge, Calligra Sheets peut aussi insérer des données d'une base de données SQL dans un tableur. Ceci se fait en utilisant l'option **Insertion** → **Données externes** → **D'une base de données...**

## 4.10 Lier les cellules

Une cellule de tableur peut être liée à une action de telle manière qu'en cliquant du bouton gauche sur la cellule s'ouvrira, par exemple, votre navigateur. Pour qu'une cellule agisse de cette manière, sélectionnez-la et choisissez **Insertion** → **Hyperlien....** Ceci ouvrira la boîte de dialogue **Créer un hyperlien**, qui vous permet de choisir entre quatre types d'hyperliens :

- Une cellule avec un lien **Internet** essayera d'ouvrir votre navigateur par défaut à l'URL donnée dans la zone de texte **Adresse Internet** de la boîte de dialogue **Insérer un lien** quand on la cliquera. Ce peut être par exemple **http://www.calligra.org**.
- En cliquant sur une cellule qui contient un hyperlien **Mail**, s'ouvre un éditeur de messages avec l'adresse écrite dans la zone de texte **Adresse électronique** : comme adresse du destinataire. Par exemple **anon@quelquepart.com**.
- Une cellule de lien vers un **Fichier** contient l'emplacement d'un fichier ou d'un dossier, comme donné dans la zone de texte **Emplacement du fichier** :, et essayera d'ouvrir ce fichier ou ce dossier avec une application adaptée lorsque cliquée.
- Le type de **Cellule** d'une cellule hyperlien contient une référence de cellule de Calligra Sheets, écrite dans la zone de texte **Cellule ou zone nommée**. Cliquer avec bouton Gauche sur ce type de cellule hyperlien fait que le focus de Calligra Sheets se déplace vers la cellule cible.

Les quatre types d'hyperliens ont besoin que du texte convenable soit écrit dans le champ **Texte à afficher** de la boîte de dialogue **Insérer un lien**. Il s'agit du texte qui apparaît dans la cellule.

## 4.11 Vérification de validité

Calligra Sheets peut vérifier automatiquement la validité de données saisies contre un certain nombre de critères, et faire surgir une boîte de dialogue si une donnée est non valable.

Pour activer cette fonction, sélectionnez les cellules à surveiller et choisissez **Données** → **Validité....** Ceci ouvre la boîte de dialogue de Calligra Sheets **Validité**, qui présente trois pages à onglets.

Dans la page **Critère**, sélectionnez quel type de donnée doit être considérée comme valable depuis la liste combinée **Autoriser** ; puis définissez la plage valable de cellules en choisissant une des options dans la liste combinée **Données** : et en donnant des valeurs adaptées dans une zone d'édition ou les deux.

Quand vous avez fait ceci, passez à l'onglet **Alerte d'erreur**. Ici, vous pouvez choisir le type de boîte de message (**Arrêter**, **Avertissement** ou **Information**) qui apparaîtra quand une valeur non valable est donnée, et définira le titre de la boîte de message et le texte du message.

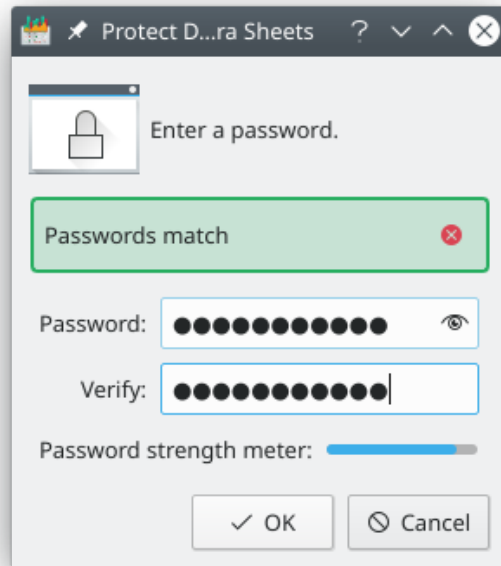
Noter que cette fonction ne vérifie que les données que vous avez saisies dans la cellule. Pour la manière de vérifier les résultats de formules, voyez la section [Attributs conditionnels des cellules](#) de ce manuel.

## 4.12 Protection

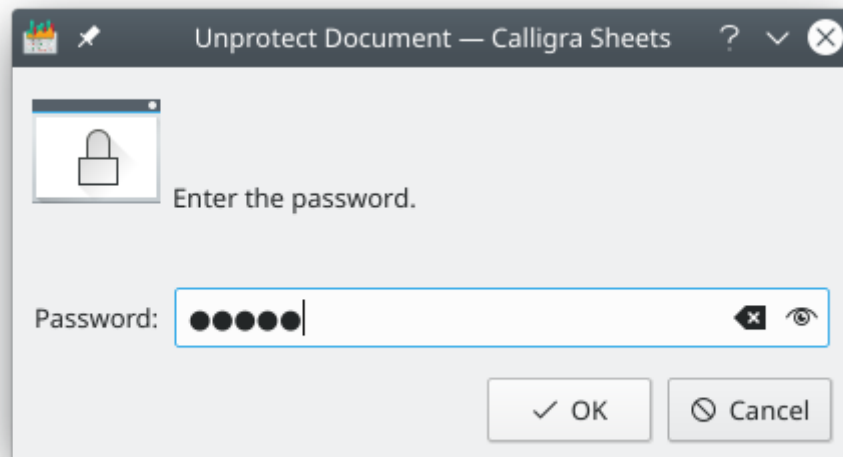
### 4.12.1 Protection de document

Protéger le document signifie que, sans mot de passe, un utilisateur ne peut ajouter ou supprimer des feuilles. La protection du document ne protège pas les cellules.

Sélectionner **Outils** → **Protéger document....** Une boîte de dialogue apparaît, vous demandant un mot de passe. L'indicateur de longueur du **Mot de passe** : vous indique si votre mot de passe est assez sécurisé. Plus il est long, plus votre mot de passe est sécurisé.



Le mot de passe sera de nouveau exigé pour déprotéger le document.



Quand un document est protégé, vous ne pouvez pas

- Renommer une feuille
- Insérer une feuille
- Supprimer une feuille
- Masquer une feuille
- Afficher une feuille
- Voir les propriétés de la feuille
- Fusionner ou dissocier les cellules



### 4.12.2 Protection de la feuille

Protéger une feuille signifie protéger le contenu de toutes les cellules protégées et des objets de la feuille. Les cellules individuelles ou une sélection de cellules peuvent être déprotégées à l'intérieur d'une feuille protégée, voir [prochaine section](#).

Pour protéger une feuille, sélectionner **Outils** → **Protéger feuille...** Une boîte de dialogue apparaît et vous demande un mot de passe. L'indicateur de longueur du **Mot de passe** vous indique si votre mot de passe est suffisamment sécurisé. Plus il est long, plus votre mot de passe est sécurisé. Cette protection de cellule est active pour toutes les cellules par défaut et est effective quand vous activez la protection de la feuille. Ainsi si vous conservez les options par défaut et si vous protégez la feuille, toutes les cellules seront protégées.

Le mot de passe sera demandé pour déprotéger la feuille

Quand une feuille est protégée, vous ne pouvez pas :

- Insérer un objet ou un diagramme
- Formater n'importe quelle cellule
- Insérer une ligne ou une colonne
- Éditer et modifier le contenu d'une cellule
- Modifier un contenu dans la feuille

#### NOTE

La protection d'une cellule est très utile pour prévenir l'effacement accidentel d'une formule.

### 4.12.3 Protection d'une cellule ou d'une sélection de cellules

#### AVERTISSEMENT

La protection est active pour toutes les cellules par défaut et devient effective quand vous activez la protection de la feuille entière. Ainsi si vous conservez les options par défaut et si vous protégez la feuille, toutes les cellules seront protégées.

Si vous souhaitez protéger seulement certaines cellules, la protection par défaut doit être désactivée pour toutes les autres cellules. Par exemple, vous pouvez souhaiter que certaines cellules acceptent des entrées d'un utilisateur, aussi vous devez décocher **Protégé** pour celles-ci et conserver la protection pour les cellules qui ne doivent pas être modifiées (comme les titres). Ainsi, vous avez trois étapes à effectuer : déprotéger toutes les cellules, sélectionner les cellules à protéger et le faire, et ensuite protéger la feuille entière.

Pour déprotéger toutes les cellules :

- Sélectionner la feuille entière avec la souris.
- Dans la barre de menu, sélectionner **Format** → **Format de cellule**
- Dans la boîte de dialogue qui apparaît, aller à l'onglet **Protection de cellule**
- Décocher **Masquer tout** et décocher **Protégé** pour supprimer la protection de toutes les cellules. Les cellules sont maintenant toutes déprotégées..

Pour protéger une suite de cellules sélectionnées ou une sélection de cellules non-contiguës :

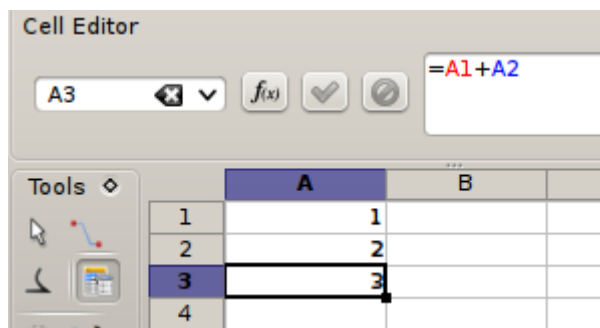
- Sélectionner l'ensemble de cellules à protéger ou utiliser la touche **Ctrl** pour sélectionner des cellules non-contiguës.
- Quand toutes les cellules souhaitées sont sélectionnées, aller à **Format** → **Format de cellule**
- Dans la boîte de dialogue qui apparaît, aller à l'onglet **Protection de cellule**
- Cliquer sur la boîte de dialogue jusqu'à **Protégé** puis cliquer sur **Ok**.

Bien que les cellules soient marquées en protection, l'option de protection doit être activée au niveau de la feuille, ce qui signifie que vous devez protéger le feuille entière pour que la cellule soit effectivement protégée.

- Sélectionner **Outils** → **Protection de feuille**.
- Dans la boîte de dialogue qui apparaît, fournir un mot de passe valable, puis le confirmer en le saisissant de nouveau. Cliquer sur **Ok**.
- Les cellules protégées elles-mêmes dans une feuille protégée ne peuvent pas être éditées sans déprotéger la feuille entière. Toutes les modifications sont désactivées. Par exemple, il est impossible d'insérer des lignes ou des colonnes; de modifier la largeur d'une colonne, ou de créer des graphiques intégrés.

#### 4.12.4 Masquer la formule de la cellule

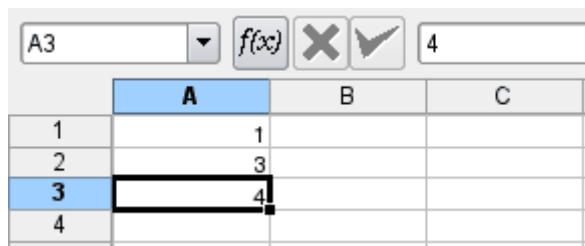
Vous pouvez souhaiter masquer votre formule pour que personne d'autre ne puisse la voir. Par défaut, chaque cellule est protégée et non masquée. Mais il est important de se souvenir que ces attributs ne sont pas effectifs tant que la feuille est protégée.



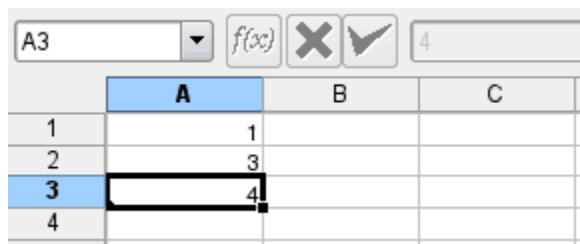
Pour masquer la formule d'une cellule, sélectionner la cellule ou la zone de cellules contiguës avec **Ctrl** et, ensuite, choisir le menu **Format** → **Format de cellule**. Dans la boîte de dialogue, cliquer sur l'onglet **Protection de cellule** et sélectionner **Masquer formule**. Après que vous ayez protégé la feuille, les résultats de la formule seront visibles, mais non la formule elle-même.

Vous devez maintenant protéger la feuille, choisir **Outils** → **Protection de feuille** pour afficher la boîte de dialogue. Saisir un mot de passe valable deux fois pour prévenir contre une suppression de la protection de la feuille.

Quand **Masquer formule** est activé et **Protégé** désactivé, la formule est masquée après la protection de la feuille mais le contenu de la cellule peut être modifié.



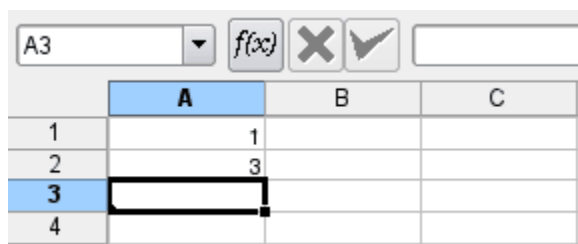
Quand **Masquer formule** et **Protégé** sont activés, la formule est masquée après la protection de la feuille et il devient impossible de modifier le contenu de la cellule.



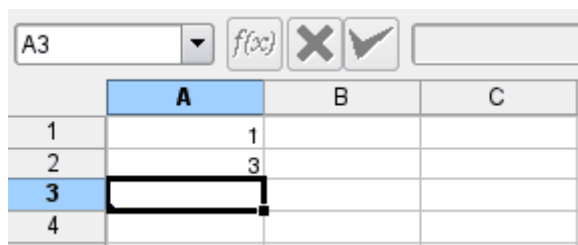
Réfléchir au fait qu'il est très facile de deviner le mot de passe d'une feuille aussi si vous recherchez une réelle sécurité, ce n'est pas la meilleure solution.

### 4.12.5 Masquer tout dans la cellule

vous pouvez changer à la fois la formule et le contenu de la cellule en choisissant **Masquer tout** dans l'onglet Protection de cellule dans le menu **Format** → **Format de cellule**.. Dans la capture d'écran ci-dessous, la cellule elle-même n'est pas protégée (**Protégé** est décoché) aussi le contenu de la cellule peut être modifié.



Là la cellule elle-même est protégée aussi elle ne peut être écrasée.



## 4.13 Autres fonctionnalités

### 4.13.1 Zones et cellules nommées

Vous pouvez donner un nom comme **foo** à une cellule ou à n'importe quelle zone d'une feuille en sélectionnant la zone puis sélectionnez **Nom de la zone...** du menu du bouton droit de la souris. Ceci ouvrira la boîte de dialogue **Nom de la zone** où vous pouvez écrire n'importe quel nom.

Vous pouvez aussi nommer une cellule ou une zone en la sélectionnant, puis en écrivant le nom dans la petite boîte de texte à l'extrémité gauche de la barre de formules, écrasant la référence de cellule qui apparaît normalement ici.

Si vous écrivez un nom qui a déjà été utilisé dans cette zone de texte, la sélection de Calligra Sheets changera pour afficher les cellules nommées.

L'option **Données** → **Zones nommées...** vous donnera une liste de noms existants et vous permettra de changer le focus de Calligra Sheets en n'importe lequel ou vous permettra de supprimer un nom.

Les cellules nommées sont particulièrement utiles dans les formules comme une alternative à **références absolues aux cellules** car les noms peuvent être utilisés à la place des références normales aux cellules et ne changent pas quand une cellule qui contient des références est copiée. Quand un nom est utilisé de cette manière, il doit être encadré d'apostrophes.

Par exemple, si la cellule A1 a pour nom **jean**, vous pouvez saisir une formule comme **= ' jean' + 2** dans une autre cellule qui doit toujours additionner la valeur de A1 sans se soucier d'où la formule a été copiée.

Notez que ces noms de cellule et de zone sont traités comme des minuscules.

### 4.13.2 Commentaires de cellules

Une cellule contient un commentaire de texte qui peut être vu en travaillant sur le tableur, mais qui n'est pas imprimé et pas affiché normalement.

Pour ajouter un commentaire, sélectionnez la cellule et choisissez **Commentaire...** du menu du bouton droit de la souris ou du menu **Insertion** et écrivez votre commentaire dans la boîte de dialogue **Commentaire de cellule** qui en résulte.

Pour voir le commentaire, passez le pointeur de la souris sur la cellule. Le commentaire apparaîtra comme si c'était une infobulle.

Si vous cochez la case **Afficher l'indicateur de commentaires** de la page **Divers** dans le dialogue **Configuration** → **Configurer KSpread...**, ces cellules qui contiennent des commentaires seront repérées avec un petit triangle rouge dans le coin supérieur droit.

Pour ouvrir cette boîte de dialogue, cliquez avec le bouton droit de la souris dans l'onglet de la fenêtre principale de la feuille et sélectionnez **Propriétés de feuille** dans le menu déroulant. Ou le sélectionner depuis le menu **Format** → **Feuille**.

Pour enlever un commentaire d'une cellule, sélectionnez **Supprimer le commentaire** du menu du bouton droit de la souris ou choisissez **Édition** → **Supprimer** → **Commentaire**.

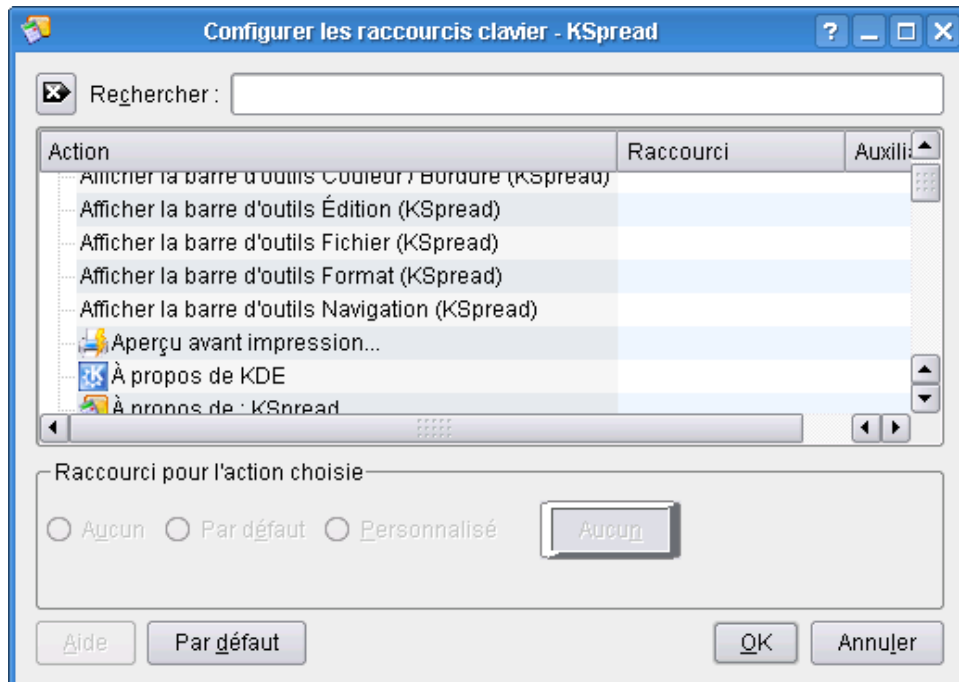
## Chapitre 5

# Configurer les raccourcis et barres d'outils de Calligra Sheets

Pamela Robert  
Traduction française : Gérard Delafond  
Traduction française : Daniel Durand  
Traduction française : Ludovic Grossard

### 5.1 Raccourcis

Pour changer les combinaisons de raccourcis clavier utilisés par Calligra Sheets, sélectionnez **Configuration** → **Configurer les raccourcis clavier...** Ceci ouvrira une boîte de dialogue comme ci-dessous.



Cherchez dans la liste déroulante pour trouver l'action dont vous voulez ajouter ou modifier le raccourci- clavier et sélectionnez-la par un clic gauche sur son nom. En donnant le nom de l'action

dans la barre de recherche en haut, vous pourrez alors changer le raccourci en sélectionnant le bouton radio **Aucun**, **Par défaut** ou **Personnalisé**.

Maintenant vous pouvez simplement actionner la combinaison de touches que vous voulez pour agir comme raccourci, par exemple **Ctrl+Maj+S**.

## 5.2 Barres d'outils

Calligra Sheets a six barres d'outils : **Fichier**, **Édition**, **Maths**, **Navigation**, **Format**, **Police** et **Couleur / Bordures**, chacune d'entre elles peut être affichée ou non en fonction des choix faits dans le menu **Configuration**.

Si les barres d'outils sont déverrouillées, vous pouvez choisir si une barre d'outils apparaît en **haut**, à **gauche**, à **droite** ou en **bas** de la fenêtre de Calligra Sheets en cliquant du bouton droit sur la barre d'outils, ce qui ouvre le **menu de barre d'outils** et en faisant une sélection dans le sous-menu **Orientation**. Ce **Menu de barre d'outils** a aussi des sous-menus pour choisir si la barre d'outils affiche des icônes, de texte ou les deux, et la taille des icônes.

Une autre manière de déplacer une barre d'outils est de placer le pointeur de la souris sur les deux barres verticales à l'extrémité gauche de chaque barre d'outils et de maintenir le bouton gauche de la souris enfoncé pendant que vous tirez la barre vers la position désirée. Quand vous tirez la barre d'outils de cette manière, vous pouvez relâcher le bouton de la souris quand il est à une certaine distance de n'importe quel bord de la fenêtre de Calligra Sheets, et vous obtiendrez alors une barre flottante, qui n'est pas liée à une partie particulière de la fenêtre de Calligra Sheets, et peut être déplacée hors de la fenêtre. Pour remettre une barre d'outils flottante dans une des positions traditionnelles, cliquez du bouton droit sur sa petite barre de titre pour amener le **Menu de barre d'outils** et choisissez une des options dans le sous-menu **Orientation**.

La sélection de **Configurer les barres d'outils...** du menu **Configuration** ouvre une boîte de dialogue qui vous permet d'ajouter des boutons ou d'en enlever des barres d'outils de Calligra Sheets.

Pour utiliser cette boîte de dialogue **Configurer les barres d'outils**, sélectionnez d'abord une barre d'outils dans la liste combinée **Barre d'outils** : La fenêtre à droite **Actions actuelles** : affichera alors les boutons actuellement présents dans la barre d'outils. Vous pouvez enlever un bouton en le sélectionnant dans cette fenêtre puis en actionnant le bouton gauche de la souris, ou le déplacer en actionnant les flèches haut et bas. Pour ajouter un nouveau bouton à la barre d'outils, sélectionnez-le dans la liste **Actions disponibles** : et actionnez le bouton en flèche vers la droite.

## Chapitre 6

# La boîte de dialogue de configuration de Calligra Sheets

Pamela Robert

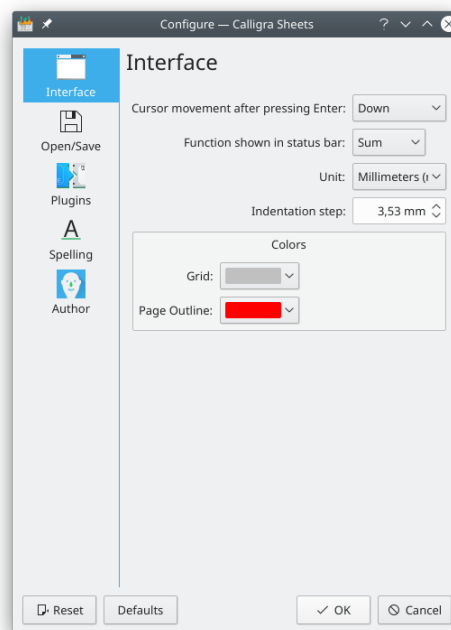
Traduction française : Gérard Delafond

Traduction française : Daniel Durand

Traduction française : Ludovic Grossard

La sélection de **Configuration** → **Configurer Calligra Sheets...** ouvre une boîte de dialogue avec plusieurs pages, sélectionnables par les icônes à gauche, qui permettent de changer plusieurs aspects des opérations de Calligra Sheets.

### 6.1 Interface



**Mouvement du curseur après la pression de Entrée :**

Quand vous avez sélectionné une cellule puis actionné la touche **Entrée**, la sélection se déplacera d'une place vers la **gauche**, la **droite**, le **haut**, le **bas**, **première colonne** ou **Aucun** comme déterminé par le réglage dans la boîte de sélection déroulante.

**Fonction montrée dans la barre d'état :**

Cette boîte à sélection peut être utilisée pour choisir le calcul fourni par la fonction [Résumé de la barre d'état](#).

**Unité :**

Choisissez l'unité par défaut à utiliser sur votre feuille.

**Pas d'indentation :**

Règle la taille de l'indentation qui sera utilisée dans une cellule lorsque vous choisissez les actions **Augmenter l'indentation** et **Diminuer l'indentation** depuis la barre d'outils. Ces actions ne sont pas activées par défaut dans la barre d'outils.

**Capte toutes les touches de navigation pendant l'édition.**

Capte toutes les touches de navigation, c'est-à-dire les touches flèches, haut/bas de page, touche de tabulation avant et arrière, pendant l'édition d'une cellule avec l'éditeur inclus. L'éditeur inclus apparaît directement dans la cellule. Si elles sont captées, ces clés sont utilisées pour la navigation dans l'éditeur. Sinon, elles sont utilisées pour la navigation de la cellule.

La section **Couleurs** vous permet de choisir la couleur du quadrillage de la feuille. Si vous ne voulez pas que le quadrillage apparaisse du tout, décochez la case **Afficher le quadrillage** dans la boîte de dialogue **Format** → **Feuille** → **Propriétés de la feuille**.

Cette section vous permet aussi de choisir la couleur des lignes utilisées pour indiquer les bordures de pages affichées quand la case **Afficher les limites de pages** dans le menu **Affichage** est cochée.

Choisissez **Personnaliser...** dans la couleur courante pour afficher la boîte de dialogue KDE standard **Choisir une couleur**.

**Grille**

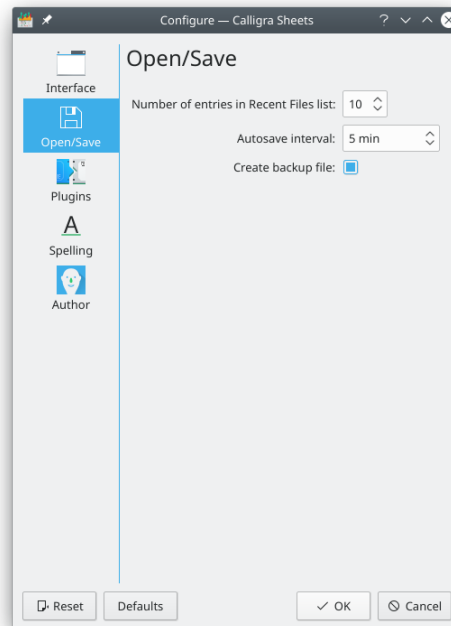
Cliquer ici pour modifier de la grille c'est-à-dire la couleur des bordures de chaque cellule

**Limites de page**

quand l'élément du menu **Vue** → **Afficher bordures de page** est coché, les bordures de page sont affichées. Cliquer ici pour choisir une autre couleur de bordure au lieu du rouge par défaut.



## 6.2 Ouvrir / Enregistrer



### Nombre de fichiers récents à afficher :

Contrôle le nombre maximal de noms de fichiers qui sont affichés quand vous sélectionnez **Fichier** → **Récemment ouverts**.

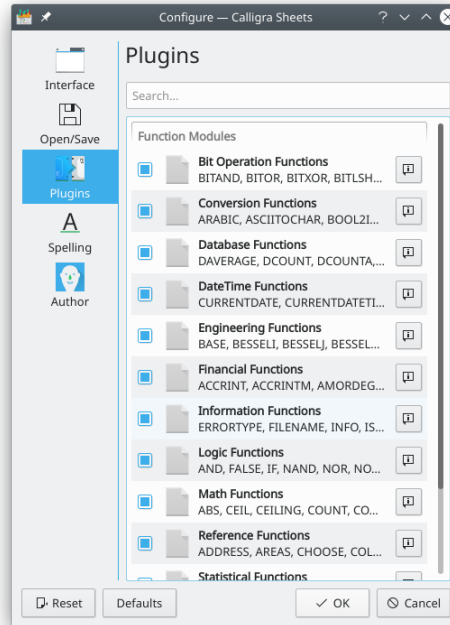
### Intervalle d'enregistrement automatique :

Vous pouvez ici sélectionner l'intervalle entre deux enregistrements automatiques, ou désactiver cette fonction en choisissant **Pas d'enregistrement automatique** (en choisissant la valeur minimale).

### Créer des fichiers de sauvegarde :

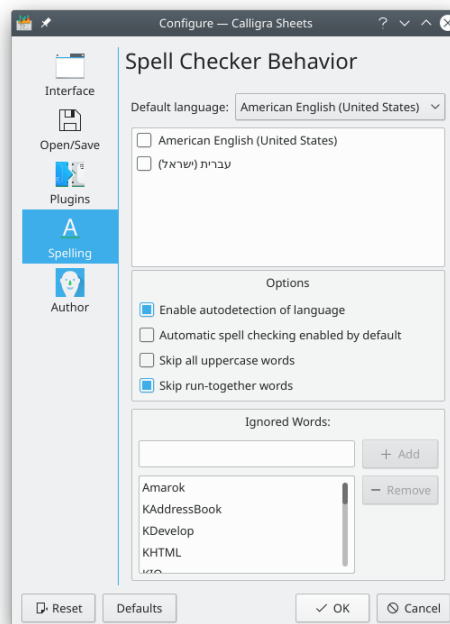
Cocher cette boîte si vous souhaitez créer des fichiers de sauvegarde. Elle est cochée par défaut.

## 6.3 Modules externes



Vérifiez un module externe dans la liste pour l'activer. Affichez la boîte de dialogue **À propos** en cliquant sur le bouton info à droite de la liste des éléments.

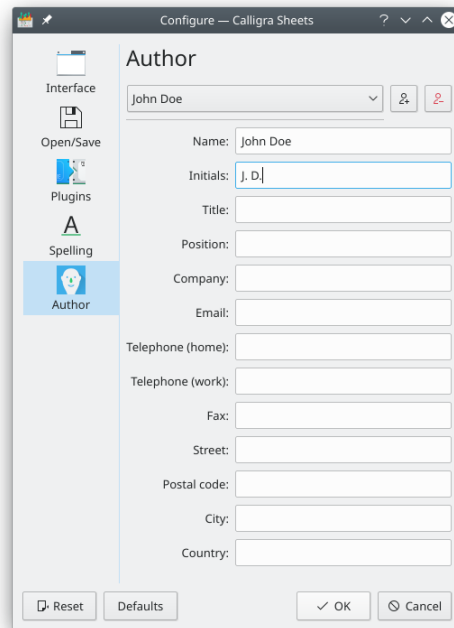
## 6.4 Vérification d'orthographe



Cette page vous permet de configurer le comportement du vérificateur d'orthographe de Calligra Sheets.

Pour plus d'informations, veuillez consulter la documentation du [correcteur orthographique](#).

## 6.5 Auteur



Choisissez un profil d'auteur pour votre document.

Vous pouvez ajouter un nouveau profil ou supprimer le profil actuel en utilisant les boutons à droite de la liste déroulante des profils.

## Chapitre 7

# Références des commandes

Pamela Robert  
Traduction française : Gérard Delafond  
Traduction française : Daniel Durand  
Traduction française : Ludovic Grossard

### 7.1 Le menu Fichier

**Fichier** → **Nouveau (Ctrl+N)**

Commencer un nouveau document

**Fichier** → **Ouvrir... (Ctrl+O)**

Ouvrir un document existant

**Fichier** → **Récemment ouverts**

Ouvre un document existant en le sélectionnant dans une liste combinée de fichiers récemment ouverts.

**Fichier** → **Enregistrer (Ctrl+S)**

Enregistrer le document

**Fichier** → **Enregistrer sous...**

Enregistre et modifie le nom et le format du document actuel. Si vous souhaitez conserver le nom et le format du document, utiliser **Exporter...**

**Fichier** → **Recharger**

Recharge le document

**Fichier** → **Importer...**

Importer d'autres documents

Les tableaux de données sont souvent contenus dans des fichiers de texte avec les valeurs d'une ligne séparées par des virgules, espaces, tabulations ou autre caractère, par exemple *123, 456, 789, abcd, efgh*. De tels fichiers sont habituellement appelés fichiers "CSV" (Comma Separated Values), même si le caractère séparateur est autre chose qu'une virgule.

Si vous demandez à Calligra Sheets d'ouvrir un fichier de texte, il suppose que ce fichier est au format CSV et lance une boîte de dialogue qui vous permet de spécifier le délimiteur (caractère de séparation) utilisé par le fichier, et affiche comment les éléments seront placés dans les différentes cellules du tableur.

Les autres options dans cette boîte de dialogue vous permettent de définir le **Format** des cellules du tableau, si les caractères guillemets doivent être supprimés et si les premières lignes du fichier doivent être ignorées.

**Fichier** → **Exporter...**

Enregistrer le document dans l'un des formats gérés. Le document ne devient pas le fichier exporté.

**Fichier → Envoyer par courrier électronique...**

Envoyer le fichier comme pièce jointe à un courrier électronique.

**Fichier → Créer un modèle à partir du document...**

Créer un [modèle](#) Calligra Sheets utilisant ce document.

**Fichier → Imprimer... (Ctrl+P)**

Imprimer le document

**Fichier → Aperçu avant impression...**

Afficher le document comme il sera imprimé.

**Fichier → Informations sur le document**

Afficher ou saisir des informations sur le document et l'auteur.

**Fichier → Fermer (Ctrl+W)**

Fermer le document courant mais en laissant Calligra Sheets en marche.

**Fichier → Quitter (Ctrl+Q)**

Quitter Calligra Sheets.

## 7.2 Le menu Modifier

**Modifier → Annuler (Ctrl+Z)**

Annuler la dernière action.

**Modifier → Refaire (Ctrl+Maj+Z)**

Refaire la dernière action annulée.

**Modifier → Couper (Ctrl+X)**

Placer les éléments sélectionnés dans le presse-papiers et les enlever de leur position d'origine. Si vous faites un **Coller**, les éléments seront inséré au nouvel endroit.

**Modifier → Copier (Ctrl+C)**

Copie les éléments sélectionnés dans le presse-papiers.

**Modifier → Coller (Ctrl+V)**

Colle les éléments du presse-papiers dans les cellules sélectionnées.

**Modifier → Collage spécial...**

Des formes spéciales de collage. Voyez les sections [Autres modes de collage](#) et [L'arithmétique en utilisant le collage spécial](#) pour plus de détails.

**Modifier → Coller avec insertion**

Déplace le contenu de la zone collée soit à droite ou en bas des éléments collés depuis le presse-papiers dans les cellules sélectionnées.

**Modifier → Remplir**

Remplis la zone sélectionnée avec les valeurs contenue dans le premier ensemble d'éléments. Les quatre directions sont possibles. Notez que le terme « ensemble d'éléments » décrit le premier ensemble de valeurs vu dans la direction de remplissage. Si la direction de remplissage est la gauche, alors le premier ensemble d'éléments est la dernière colonne de la sélection.

**Modifier → Chercher... (Ctrl+F)**

Trouver une cellule qui contient du texte.

**Modifier → Poursuivre la recherche (F3)**

Trouve la cellule suivante contenant le texte donné.

**Modifier → Chercher (Maj+F)**

Trouve la cellule précédente contenant le texte donné.

**Modifier → Remplacer... (Ctrl+R)**

Chercher et remplacer le texte donné dans les cellules.

**Modifier** → **Effacer**

Efface **Tout** ou le **contenu**, les **commentaires**, les **styles conditionnels**, les **liens** ou la **validité** des cellules sélectionnées.

**Modifier** → **Supprimer**

Supprime les **cellules**, les **colonnes**, les **lignes** ou la **feuille**.

**Modifier** → **Poursuivre la recherche (F3)**

Pour modifier la cellule sélectionnée in situ.

## 7.3 Le menu Vue

**Vue** → **Nouvelle vue**

Ouvrir une nouvelle instance de Calligra Sheets avec le même document.

**Vue** → **Limites de page**

Basculer en mode affichage des bordures d'impression rouges ou non.

**Vue** → **Zoom**

Augmenter ou diminuer le grossissement utilisé pour afficher le tableur. Amplitude de 33 % à 500 %.

## 7.4 Le menu Aller

Vous trouverez dans ce menu des actions permettant de naviguer entre les feuilles du document actuellement ouvert, ainsi que l'action **Aller à la cellule** pour se déplacer dans une cellule unique ou pour sélectionner une plage de cellules.

## 7.5 Le menu Insertion

**Insérer** → **Commentaire**

Ajoute ou modifie un commentaire.

**Insertion** → **Fonction...**

Insérer une fonction mathématique. Voyez la section [formules](#) pour plus de détails.

**Insertion** → **Série...**

Insérer une série. Voyez la section [Séries](#) pour plus de détails.

**Insertion** → **Hyperlien...**

Insérer un hyperlien. dans la cellule sélectionnée. Voyez la section [Cellules hyperliens](#) pour plus de détails.

**Insertion** → **Caractère spécial...**

Ajoute un caractère spécial dans la cellule sélectionnée.

**Insérer** → **Données externes**

Insérer des données à **partir d'un fichier de texte**, **une base de données** ou à **partir du presse-papiers**. Voir la section [Insertion de données externes](#) pour plus de détails.

## 7.6 Le menu Format

**Format** → **Format de cellule... (Alt+Ctrl+F)**

Formate les cellules sélectionnées. Voyez la section [Formatage du tableur](#) pour plus de détails.

**Format → Gestionnaire de style**

Crée, modifie ou supprime les styles de formatage des cellules.

**Format → Style**

Applique un style aux cellules sélectionnées. Pour gérer les styles, utilisez **Format → Gestionnaire de styles...**

**Format → Créer un style à partir de la cellule...**

Crée un nouveau style à partir du format de la cellule sélectionnée. Pour gérer les styles, utilisez **Format → Gestionnaire de styles...**

**Format → Formater automatiquement**

Formater automatiquement les cellules sélectionnées : une boîte de dialogue vous laisse sélectionner entre deux propositions de format.

**Format → Fusionner les cellules**

Fusionner les cellules sélectionnées.

**Format → Dissocier les cellules**

Dissocier les cellules précédemment fusionnées.

**Format → Ajuster la ligne et la colonne**

Déterminer la taille des lignes et des colonnes pour afficher les cellules sélectionnées correctement.

**Format → Ligne**

Redimensionner, égaliser, cacher ou afficher les lignes.

**Format → Colonne**

Redimensionner, égaliser, cacher ou afficher les colonnes.

**Format → Feuille**

Masque, montre les feuilles ou configure les propriétés expertes des feuilles.

**Format → Format de la page...**

Format de la mise en page d'impression.

**Format → Plage imprimable**

Définir ou réinitialiser la plage à imprimer.

## 7.7 Le menu Données

**Données → Trier...**

Trier les données dans les cellules sélectionnées. Voyez la section [Tri des données](#) pour plus de détails.

**Données → De texte en colonnes...**

Cette option tente d'interpréter le texte dans les cellules sélectionnées comme des données CVS, plaçant chaque élément dans une cellule différente de la ligne.

**Données → Nommer des zones... (Ctrl+Maj+G)**

Ouvre la boîte de dialogue **Zones nommées...** pour choisir, ajouter, modifier et supprimer des zones nommées. Voyez la section [Cellules et zones nommées](#) pour plus de détails.

**Données → Consolider...**

Consolider les données. Voyez la section [Consolider les données](#) pour plus de détails.

**Données → Sous-totaux...**

Crée différents types de sous-totaux sur un ensemble de données.

**Données → Validité...**

Déterminer ou modifier les critères de vérification des erreurs et des messages d'alerte pour les cellules sélectionnées. Voir [Contrôle de validité](#) pour plus de détails.

**Données → Recherche de solution...**

Ouvre la boîte de dialogue de recherche de solution. Voir [Recherche de solution](#) pour les détails.

**Données → Pivot...**

Ouvre la boîte de dialogue de recherche de configuration des tables de pivot. Voir [les tables de pivot](#) pour les détails.

## 7.8 Le menu Outils

### Outils → Vérification d'orthographe...

Vérifier l'orthographe des mots dans la feuille.

### Outils → Listes personnalisées...

Afficher ou modifier les séries spéciales de mots reconnus par Calligra Sheets. Ces listes peuvent être utilisées pour insérer des séries spéciales dans la feuille.

### Outils → Protection de feuille...

Protéger la feuille avec un mot de passe. Une boîte de dialogue vous demande un mot de passe. Décocher cette option vous demandera ce mot de passe pour déprotéger la feuille. La protection d'une feuille signifie la protection de toutes les cellules de la feuille. Dans une feuille protégée, les cellules ne peuvent pas être reformatées ou écrasées.

### Outils → Protection de documents...

Protéger le document complet avec un mot de passe. Une boîte de dialogue vous demande un mot de passe. Décocher cette option vous demandera ce mot de passe pour déprotéger le document. Il est impossible de renommer ou déplacer une feuille dans un document protégé. La protection d'un document ne signifie pas que toutes les feuilles individuelles soient protégées.

### Outils → Recalculer la feuille (Maj+F9)

Recalculer les formules dans la feuille courante.

### Outils → Recalculer le livret (F9)

Recalculer toutes les feuilles.

### Outils → Exécuter un fichier de script

Exécute un script externe dans Calligra Sheets. Calligra Sheets gère les scripts en JavaScript, Python et Ruby. Des exemples de scripts peuvent être trouvés dans le sous-menu **Outils** → **Scripts**.

### Outils → Scripts

Vous pouvez ici exécuter des scripts pour exporter ou importer des données dans différents formats, enregistrer vos logs de Calligra Sheets dans un fichier, utiliser [Orca speech](#) pour l'accessibilité, déboguer des scripts Python et Ruby, ou ajouter des fonctions pour afficher des données de stocks ou les conditions météo. Vous pouvez même utiliser des [fonctions R](#) (le module [RPy](#) doit être installé).

### Outils → Gestionnaire de scripts...

Ouvrir la boîte de dialogue **Manager de scripts** pour exécuter, charger, décharger, installer, désinstaller les scripts.

### Outils → Optimiseur de fonctions...

Ouvre la boîte de dialogue **Optimiseur de fonctions** dans laquelle vous pouvez choisir une cellule fonction objectif, une cible d'optimisation (**Maximiser**, **Minimiser** ou saisir la **valeur**) et un ensemble de cellules de paramètres de décision.

## 7.9 Le menu Configuration

### Paramètres → Barres d'outils affichées

Afficher ou masquer les barres d'outils : **Fichier**, **Éditer**, **Navigation**, **Format**, **Police** et **Couleur/Bordure**.

### Paramètres → Barre d'état

Affiche ou masque la barre d'état. Cette barre affiche des informations supplémentaires pour les éléments sélectionnée et des calculs immédiats sur les cellules sélectionnées.

### Paramètres → Barre d'onglets

Afficher ou masquer la barre d'onglets. Vous pouvez accéder grâce à cette barre à toutes les feuilles du document actuel.



**Paramètres → Configurer les raccourcis claviers..**

Configurer les raccourcis clavier utilisés par Calligra Sheets. Voyez la section [Configurer les raccourcis](#) pour plus de détails.

**Paramètres → Configurer les barres d’outils...**

Configurer les barres d’outils. La section sur [la configuration des barres d’outils](#) possède plus d’informations.

**Paramètres → Thèmes**

Choisit le thème de couleur pour la fenêtre de Calligra Sheets. Vous pouvez choisir l’un des modèles de couleurs prédéfinis, ou sélectionner **Configuration...** pour ouvrir le [module de sélection de couleurs de la configuration du système](#).

**Paramètres → Profil de l’auteur actif**

Configure le profil de l’auteur pour le document en cours. Vous pouvez choisir un des profils définis en utilisant [la fenêtre de configuration de Calligra Sheets](#), le **profil par défaut de l’auteur** défini dans la configuration du système ou un profil vise **anonyme** qui protège votre vie privée.

**Paramètres → Configurer les notifications**

Configure le système de notifications de Calligra Sheets. Il n’y a aucune action qui pourrait vous être notifiée dans la version actuelle de Calligra Sheets.

**Paramètres → Configurer Calligra Sheets...**

La configuration générale de Calligra Sheets. Voyez la section sur [la configuration de Calligra Sheets](#) pour plus de détails.

## 7.10 Le menu Aide

**Aide → Manuel utilisateur de Calligra Sheets (F1)**

Exécute le système d’aide de KDE en commençant par les pages d’aide de Calligra Sheets (ce document).

**Aide → Qu’est-ce que c’est? (Maj+F1)**

Change le curseur de la souris en une combinaison flèche et point d’interrogation (?). Cliquer sur des éléments dans Calligra Sheets ouvrira une fenêtre d’aide (s’il en existe une pour cet élément particulier), expliquant la fonction de l’élément en question.

**Aide → Rapport de bogue...**

Ouvre une boîte de dialogue de rapport de bogue où vous pouvez signaler un bogue ou effectuer une requête de fonctionnalité sous la forme d’une liste de souhaits (“wishlist”).

**Aide → À propos de Calligra Sheets**

Affiche des informations sur la version et sur l’auteur.

**Aide → À propos de KDE**

Affiche la version de KDE et d’autres informations de base.

## 7.11 Le menu du bouton droit de la souris

Cette section décrit les éléments du menu contextuel obtenu en cliquant du bouton droit sur des cellules, lignes ou colonnes sélectionnées.

**Format de cellule... (Ctrl+Alt+F)**

Formate les cellules sélectionnées. Voyez la section [Formatage du tableur](#) pour plus de détails.

**Couper (Ctrl+X)**

Met les éléments sélectionnés dans le presse-papiers. Si vous faites alors un **Coller**, les éléments seront déplacés de l’emplacement d’origine vers le nouveau.

**Copier (Ctrl+C)**

Copier les éléments sélectionnés dans le presse-papiers.

**Coller (Ctrl+V)**

Coller les éléments du presse-papiers dans les cellules sélectionnées.

**Collage spécial...**

Des formes spéciales de collage. Voyez les sections [Autres modes de collage](#) et [L'arithmétique en utilisant le collage spécial](#) pour plus de détails.

**Coller avec insertion**

Coller du presse-papiers vers les cellules sélectionnées, en déplaçant les cellules précédentes pour faire de la place.

**Toutes**

Supprimer le contenu des cellules sélectionnées.

**Ajuster les lignes & les colonnes**

Changer la taille des lignes et des colonnes pour afficher les cellules sélectionnées complètement.

**Par défaut**

Détermine les formats par défaut pour les cellules sélectionnées.

**Nom de la zone...**

Nommer les zones sélectionnées. Voyez la section [Zones nommées](#) pour plus de détails.

**Redimensionner la ligne...**

Changer la hauteur de la ligne sélectionnée.

**Ajuster la ligne**

Changer la hauteur de la ligne sélectionnée pour afficher complètement les cellules.

**Redimensionner la colonne...**

Changer la largeur de la colonne sélectionnée.

**Ajuster la colonne**

Changer la largeur de la colonne sélectionnée pour afficher complètement les cellules.

**Insérer cellule(s)...**

Insérer de nouvelles cellules à l'emplacement sélectionné, déplaçant les cellules existantes pour faire de la place.

**Supprimer des cellules...**

Supprimer les cellules sélectionnées, déplaçant les autres cellules pour occuper l'espace laissé libre par les cellules supprimées.

**Insérer ligne(s)...**

Insérer de nouvelles lignes au-dessus des lignes sélectionnées.

**Supprimer lignes**

Supprimer les lignes sélectionnées.

**Masquer les rangées**

Cache les lignes sélectionnées.

**Afficher les rangées**

Affiche les lignes sélectionnées. Pour afficher les lignes cachées, vous avez besoin de sélectionner une plage de lignes qui inclut les lignes cachées.

**Insérer colonnes**

Insérer de nouvelles colonnes à gauche des colonnes sélectionnées.

**Supprimer colonnes**

Supprimer les colonnes sélectionnées.

**Masquer colonnes**

Cache les colonnes sélectionnées.

**Afficher les colonnes**

Affiche les colonnes sélectionnées. Pour afficher les colonnes cachées, vous avez besoin de sélectionner une plage de colonnes qui inclut les colonnes cachées.

**Commentaire...**

Ajoute ou modifie un commentaire aux cellules sélectionnées.

**Liste de sélection...**

Vous permet de sélectionner et coller le texte de n'importe quelle autre cellule dans la cellule sélectionnée.

## 7.12 Autres raccourcis

Cette section décrit les raccourcis clavier de Calligra Sheets utilisés pour les opérations qui n'apparaissent pas dans les menus.

**Ctrl+H**

(Dés)active l'affichage des dockers.

**Ctrl+Touches Flèches**

Si la cellule sélectionnée est occupée, déplacer la sélection au début ou à la fin des blocs occupés dans la ligne ou colonne courante. Si la cellule sélectionnée n'est pas occupée, déplacer la sélection au début ou à la fin du bloc de cellules inoccupées dans la ligne ou colonne courante.

**Ctrl+Maj+Touches flèches**

Si la cellule sélectionnée est occupée, sélectionne toutes les cellules occupées au début ou à la fin de ce bloc de cellules occupées dans la ligne ou colonne courante. Si la cellule sélectionnée n'est pas occupée, sélectionne toutes les cellules occupées au début ou à la fin de ce bloc de cellules inoccupées dans la ligne ou colonne courante.

**Bas de page**

Déplacer le curseur de cellules de 10 cellules vers le bas.

**Haut de page**

Déplacer le curseur de cellules de 10 cellules vers le haut.

**Ctrl+Bas de page**

Déplacer vers la feuille suivante.

**Ctrl+Haut de page**

Aller à la feuille précédente.

**F4**

Changer la référence de cellule entre les types normal et [référence absolue](#).

## Chapitre 8

# Fonctions

Calligra Sheets dispose d'un grand nombre de fonctions mathématiques intégrées et d'autres fonctions qui peuvent être utilisées dans des formules dans des cellules.

### 8.1 Fonctions prises en charge

Ce chapitre est un bref aperçu de toutes les fonctions prises en charge dans les groupes suivants :

- Opérations par bit
- Conversion
- Base de données
- Date et heure
- Technique
- Financières
- Information
- Logiques
- Recherche et références
- Mathématiques
- Statistiques
- Texte
- Trigonométrie

#### 8.1.1 Opérations par bit

##### 8.1.1.1 BITAND

La fonction BITAND() produit une opération AND bit à bit pour les deux paramètres intégrés.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

BITAND(valeur;valeur)

##### Paramètres

*Commentaire* : Premier nombre, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Deuxième nombre, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

BITAND(12;10) renvoie 8 (car la décimale 12 est une binaire de 1100, et la décimale 10 est une binaire de 1010; et 1100 « anded » avec 1010 est 1000, qui est un entier de 8).

### Fonctions en relation

[BITOR](#)  
[BITXOR](#)

#### 8.1.1.2 BITLSHIFT

La fonction BITLSHIFT() produit un déplacement par unités de l'ascenseur à gauche selon le premier paramètre. Le nombre d'unités du déplacement est spécifié par le second paramètre. Noter qu'un nombre négatif d'unités à déplacer à gauche produit un déplacement à droite.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

#### Syntaxe

BITLSHIFT(valeur ; taille de déplacement)

#### Paramètres

*Commentaire* : Premier nombre, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Monter l'ascenseur de gauche par, *Type* : Un nombre entier (comme 1, 132, 2344)

### Fonctions en relation

[BITLSHIFT](#)

#### 8.1.1.3 BITOR

La fonction BITOR() produit une opération OR par unités pour les deux paramètres intégrés.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

#### Syntaxe

BITOR(valeur ; valeur)

#### Paramètres

*Commentaire* : Premier nombre, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Deuxième nombre, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

BITOR(12;10) renvoie 14 (car la décimale de 12 est une binaire de 1100, et une décimale de 10 est une binaire de 1010 ; et 1100 « ored » avec 1010 est 1110, qui est un entier de 14).

### Fonctions en relation

[BITAND](#)  
[BITXOR](#)

#### 8.1.1.4 BITRSHIFT

La fonction BITRSHIFT() produit un déplacement par unités de l'ascenseur à droite selon le premier paramètre. Le nombre d'unités du déplacement est spécifié par le second paramètre. Noter qu'un nombre négatif d'unités à déplacer à droite produit un déplacement à gauche.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

#### Syntaxe

BITRSHIFT(valeur ; taille de déplacement)

#### Paramètres

*Commentaire* : Premier nombre, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Monter l'ascenseur de droite par, *Type* : Un nombre entier (comme 1, 132, 2344)

### Fonctions en relation

[BITLSHIFT](#)

### 8.1.1.5 BITXOR

La fonction BITXOR() produit une opération OR exclusivement par unités pour les deux paramètres intégrés.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

#### Syntaxe

BITXOR(valeur;valeur;...)

#### Paramètres

*Commentaire* : Premier nombre, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Deuxième nombre, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

BITXOR(12;10) renvoie 6 (car la décimale 12 est une binaire de 1100, et la décimale 10 est une binaire de 1010; et 1100 « xored » avec 1010 est 0110, qui est un entier de 6).

#### Fonctions en relation

[BITAND](#)

[BITOR](#)

## 8.1.2 Conversion

### 8.1.2.1 ARABIC

La fonction ARABIC() convertit un nombre romain en un nombre.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

#### Syntaxe

ARABIC(Nombre)

#### Paramètres

*Commentaire* : Numérique, *Type* : Texte

#### Exemples

ARABIC("IV") renvoie 4

#### Exemples

ARABIC("XCIX") renvoie 99

#### Fonctions en relation

[ROMAN](#)

### 8.1.2.2 ASCIITOCHAR

La fonction ASCIITOCHAR() renvoie le caractère pour chaque code ASCII donné

*Type de retour* : Texte

#### Syntaxe

ASCIITOCHAR(valeur)

#### Paramètres

*Commentaire* : La valeur ASCII à convertir, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

ASCIITOCHAR(118) renvoie "v"

#### Exemples

ASCIITOCHAR(75; 68; 69) renvoie "KDE"

### 8.1.2.3 BOOL2INT

La fonction BOOL2INT() renvoie une valeur entière pour une valeur booléenne donnée. Cette méthode est prévue pour utiliser une valeur booléenne dans les méthodes qui ont besoin d'un entier.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

#### Syntaxe

BOOL2INT(valeur)

#### Paramètres

*Commentaire* : Valeur booléenne à convertir, *Type* : Une valeur de vérité (VRAI ou FAUX)

#### Exemples

BOOL2INT(Vrai) renvoie 1

#### Exemples

BOOL2INT(Faux) renvoie 0

#### Fonctions en relation

[INT2BOOL](#)

### 8.1.2.4 BOOL2STRING

La fonction BOOL2STRING() renvoie une valeur de chaîne pour une valeur booléenne donnée. Cette méthode est prévue pour utiliser un booléen dans les méthodes qui ont besoin d'une chaîne.

*Type de retour* : Texte

#### Syntaxe

BOOL2STRING(valeur)

#### Paramètres

*Commentaire* : Valeur booléenne à convertir, *Type* : Une valeur de vérité (VRAI ou FAUX)

#### Exemples

BOOL2STRING(vrai) renvoie "Vrai"

#### Exemples

BOOL2STRING(faux) renvoie "Faux"

#### Exemples

upper(BOOL2STRING(find("nan";"banana"))) renvoie VRAI

### 8.1.2.5 CARX

La fonction CARX() renvoie l'abscisse correspondant à la position d'un point dans un espace polaire.

*Type de retour* : Double

#### Syntaxe

CARX(Rayon;Angle)

#### Paramètres

*Commentaire* : Rayon, *Type* : Double

*Commentaire* : Angle (radians), *Type* : Double

#### Exemples

CARX(12;1,5707) renvoie 0,00115592

#### Exemples

CARX(12;0) renvoie 12

#### Fonctions en relation

[CARY](#)

[POLA](#)

[POLR](#)

#### 8.1.2.6 CARY

La fonction CARY() renvoie l'ordonnée correspondant à la position d'un point dans un espace polaire.

*Type de retour* : Double

##### Syntaxe

CARY(Rayon;Angle)

##### Paramètres

*Commentaire* : Rayon, *Type* : Double

*Commentaire* : Angle (radians), *Type* : Double

##### Exemples

CARY(12;1.5707) renvoie 12

##### Exemples

CARY(12;0) renvoie 0

##### Fonctions en relation

[CARX](#)

[POLA](#)

[POLR](#)

#### 8.1.2.7 CHARTOASCII

La fonction CHARTOASCII() renvoie le code ASCII pour le caractère donné.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

CHARTOASCII(valeur)

##### Paramètres

*Commentaire* : Une chaîne d'un caractère à convertir, *Type* : Texte

##### Exemples

CHARTOASCII("v") renvoie 118

##### Exemples

CHARTOASCII(r) est une erreur. Le caractère doit être entre guillemets.

#### 8.1.2.8 DECSEX

La fonction DECSEX() convertit une valeur double en valeur de temps.

*Type de retour* : Double

##### Syntaxe

DECSEX(double)

##### Paramètres

*Commentaire* : Valeur, *Type* : Double

##### Exemples

DECSEX(1.6668) renvoie 1 :40

##### Exemples

DECSEX(7,8) renvoie 7 :47



### 8.1.2.9 INT2BOOL

La fonction INT2BOOL() renvoie une valeur booléenne pour un entier donné. Cette méthode est prévue pour utiliser un entier dans les méthodes qui ont besoin d'un booléen. Elle n'accepte que 0 ou 1. Si toute autre valeur est donnée, il est retourné Faux.

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

#### Syntaxe

INT2BOOL(valeur)

#### Paramètres

*Commentaire* : Valeur entière à convertir, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

INT2BOOL(1) renvoie vrai

#### Exemples

INT2BOOL(0) renvoie Faux

#### Exemples

OR(INT2BOOL(1), faux) renvoie vrai

#### Fonctions en relation

[BOOL2INT](#)

### 8.1.2.10 NUM2STRING

La fonction NUM2STRING() renvoie une valeur de chaîne pour un nombre donné. Notez que Calligra Sheets peut convertir automatiquement les nombres en chaîne si nécessaire; cette fonction est par conséquent rarement requise.

*Type de retour* : Texte

#### Syntaxe

NUM2STRING(valeur)

#### Paramètres

*Commentaire* : Nombre à convertir en chaîne, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

NUM2STRING(10) renvoie "10"

#### Exemples

NUM2STRING(2,05) renvoie "2,05"

#### Exemples

=find("101";NUM2STRING(A1)) (A1 = 2.010102) renvoie Vrai

#### Fonctions en relation

[STRING](#)

### 8.1.2.11 POLA

La fonction POLA() renvoie l'angle (en radians) correspondant à la position d'un point dans un espace cartésien.

*Type de retour* : Double

#### Syntaxe

POLA(X;Y)

**Paramètres**

*Commentaire* : Valeur en X, *Type* : Double

*Commentaire* : Valeur en Y, *Type* : Double

**Exemples**

POLA(12;12) renvoie 0,78539816

**Exemples**

POLA(12;0) renvoie 0

**Exemples**

POLA(0;12) renvoie 1,5707

**Fonctions en relation**

[POLR](#)

[CARX](#)

[CARY](#)

### 8.1.2.12 POLR

La fonction POLR() renvoie le rayon correspondant à la position d'un point dans un espace cartésien.

*Type de retour* : Double

**Syntaxe**

POLR(X;Y)

**Paramètres**

*Commentaire* : Valeur en X, *Type* : Double

*Commentaire* : Valeur en Y, *Type* : Double

**Exemples**

POLR(12;12) renvoie 16,9705

**Exemples**

POLR(12;0) renvoie 12

**Fonctions en relation**

[POLA](#)

[CARX](#)

[CARY](#)

### 8.1.2.13 ROMAN

La fonction ROMAN() renvoie le nombre en chiffre romain. Le nombre doit être entier et positif. L'argument optionnel du Format le niveau de concision et la valeur par défaut est 0.

*Type de retour* : Texte

**Syntaxe**

ROMAN(Nombre)

**Paramètres**

*Commentaire* : Nombre, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Format, *Type* : Un nombre entier (comme 1, 132, 2344)

**Exemples**

ROMAN(99) renvoie "XCIX"

**Exemples**

ROMAN(-55) renvoie "Err"

**Fonctions en relation**

[ARABIC](#)

#### 8.1.2.14 SEXDEC

La fonction SEXDEC() renvoie une valeur décimale. Vous pouvez aussi donner une valeur de temps.

*Type de retour* : Double

##### Syntaxe

SEXDEC(valeur de temps) ou SEXDEC(heures;minutes;secondes)

##### Paramètres

*Commentaire* : Heures, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Minutes, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Secondes, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

SEXDEC(1;5;7) renvoie 1,0852778

##### Exemples

SEXDEC("8 :05") renvoie 8,08333333

#### 8.1.2.15 CHAÎNE

La fonction STRING() renvoie une valeur de chaîne pour un nombre donné. Elle est identique à la fonction NUM2STRING().

*Type de retour* : Texte

##### Syntaxe

##### Paramètres

*Commentaire* : Nombre à convertir en chaîne, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Fonctions en relation

[NUM2STRING](#)

### 8.1.3 Base de données

#### 8.1.3.1 DAVERAGE

Calcule la moyenne dans une colonne d'une base de données spécifiée par un jeu de conditions pour les valeurs qui sont des nombres

*Type de retour* : FLOAT

##### Syntaxe

DAVERAGE(Base\_de\_données; "en-tête"; Conditions)

##### Paramètres

*Commentaire* : Plage marquant la base de données, *Type* : Une plage de chaînes

*Commentaire* : Chaîne marquant la colonne dans la base de données, *Type* : Texte

*Commentaire* : Plage marquant les conditions, *Type* : Une plage de chaînes

##### Exemples

DAVERAGE(A1 :C5; "Salaire"; A9 :A11)

### 8.1.3.2 DCOUNT

Compte les cellules qui contiennent des valeurs numériques dans une colonne d'une base de données spécifiée par un jeu de conditions.

*Type de retour* : FLOAT

#### Syntaxe

DCOUNT(Base\_de\_données; "en-tête"; Conditions)

#### Paramètres

*Commentaire* : Plage marquant la base de données, *Type* : Une plage de chaînes

*Commentaire* : Chaîne marquant la colonne dans la base de données, *Type* : Texte

*Commentaire* : Plage marquant les conditions, *Type* : Une plage de chaînes

#### Exemples

DCOUNT(A1 :C5; "Salaire"; A9 :A11)

#### Fonctions en relation

[DCOUNTA](#)

### 8.1.3.3 DCOUNTA

Compte les cellules qui contiennent des valeurs numériques ou alphanumériques dans une colonne d'une base de données spécifiée par un jeu de conditions.

*Type de retour* : FLOAT

#### Syntaxe

DCOUNTA(Base\_de\_données; "en-tête"; Conditions)

#### Paramètres

*Commentaire* : Plage marquant la base de données, *Type* : Une plage de chaînes

*Commentaire* : Chaîne marquant la colonne dans la base de données, *Type* : Texte

*Commentaire* : Plage marquant les conditions, *Type* : Une plage de chaînes

#### Exemples

DCOUNTA(A1 :C5; "Salaire"; A9 :A11)

#### Fonctions en relation

[DCOUNT](#)

### 8.1.3.4 DGET

Renvoie une simple valeur d'une colonne d'une base de données spécifiée par un jeu de conditions. Cette fonction renvoie une erreur si aucune valeur ou plus d'une valeur existent.

*Type de retour* : FLOAT

#### Syntaxe

DGET(Base\_de\_données; "en-tête"; Conditions)

#### Paramètres

*Commentaire* : Plage marquant la base de données, *Type* : Une plage de chaînes

*Commentaire* : Chaîne marquant la colonne dans la base de données, *Type* : Texte

*Commentaire* : Plage marquant les conditions, *Type* : Une plage de chaînes

#### Exemples

DGET(A1 :C5; "Salaire"; A9 :A11)

### 8.1.3.5 DMAX

renvoie la plus grande valeur dans une colonne d'une base de données spécifiée par un jeu de conditions.

*Type de retour* : FLOAT

#### Syntaxe

DMAX(Base\_de\_données; "en-tête"; Conditions)

#### Paramètres

*Commentaire* : Plage marquant la base de données, *Type* : Une plage de chaînes

*Commentaire* : Chaîne marquant la colonne dans la base de données, *Type* : Texte

*Commentaire* : Plage marquant les conditions, *Type* : Une plage de chaînes

#### Exemples

DMAX(A1 :C5; "Salaire"; A9 :A11)

#### Fonctions en relation

[DMIN](#)

### 8.1.3.6 DMIN

renvoie la plus petite valeur dans une colonne d'une base de données spécifiée par un jeu de conditions.

*Type de retour* : FLOAT

#### Syntaxe

DMIN(Base\_de\_données; "en-tête"; Conditions)

#### Paramètres

*Commentaire* : Plage marquant la base de données, *Type* : Une plage de chaînes

*Commentaire* : Chaîne marquant la colonne dans la base de données, *Type* : Texte

*Commentaire* : Plage marquant les conditions, *Type* : Une plage de chaînes

#### Exemples

DMIN(A1 :C5; "Salaire"; A9 :A11)

#### Fonctions en relation

[DMAX](#)

### 8.1.3.7 DPRODUCT

renvoie le produit de toutes les valeurs numériques dans une colonne de base de données spécifiée par un jeu de conditions.

*Type de retour* : FLOAT

#### Syntaxe

DPRODUCT(Base\_de\_données; "en-tête"; Conditions)

#### Paramètres

*Commentaire* : Plage marquant la base de données, *Type* : Une plage de chaînes

*Commentaire* : Chaîne marquant la colonne dans la base de données, *Type* : Texte

*Commentaire* : Plage marquant les conditions, *Type* : Une plage de chaînes

#### Exemples

DPRODUCT(A1 :C5; "Salaire"; A9 :A11)

#### 8.1.3.8 DSTDEV

renvoie l'estimation de l'écart-type d'une population en fonction d'un échantillon qui utilise toutes les valeurs numériques dans une colonne d'une base de données spécifiée par un jeu de conditions.

*Type de retour* : FLOAT

##### Syntaxe

DSTDEV(Base\_de\_données; "en-tête"; Conditions)

##### Paramètres

*Commentaire* : Plage marquant la base de données, *Type* : Une plage de chaînes

*Commentaire* : Chaîne marquant la colonne dans la base de données, *Type* : Texte

*Commentaire* : Plage marquant les conditions, *Type* : Une plage de chaînes

##### Exemples

DSTDEV(A1 :C5; "Salaire"; A9 :A11)

##### Fonctions en relation

[DSTDEVP](#)

#### 8.1.3.9 DSTDEVP

renvoie l'estimation de l'écart-type d'une population entière en utilisant toutes les valeurs numériques dans une colonne d'une base de données spécifiée par un jeu de conditions.

*Type de retour* : FLOAT

##### Syntaxe

DSTDEVP(Base\_de\_données; "en-tête"; Conditions)

##### Paramètres

*Commentaire* : Plage marquant la base de données, *Type* : Une plage de chaînes

*Commentaire* : Chaîne marquant la colonne dans la base de données, *Type* : Texte

*Commentaire* : Plage marquant les conditions, *Type* : Une plage de chaînes

##### Exemples

DSTDEVP(A1 :C5; "Salaire"; A9 :A11)

##### Fonctions en relation

[DSTDEV](#)

#### 8.1.3.10 DSUM

Additionne les nombres dans une colonne d'une base de données spécifiée par un jeu de conditions.

*Type de retour* : FLOAT

##### Syntaxe

DSUM(Base\_de\_données; "en-tête"; Conditions)

##### Paramètres

*Commentaire* : Plage marquant la base de données, *Type* : Une plage de chaînes

*Commentaire* : Chaîne marquant la colonne dans la base de données, *Type* : Texte

*Commentaire* : Plage marquant les conditions, *Type* : Une plage de chaînes

##### Exemples

DSUM(A1 :C5; "Salaire"; A9 :A11)

#### 8.1.3.11 DVAR

renvoie l'estimation de la variance d'une population fondée sur un échantillon qui utilise toutes les valeurs numériques dans une colonne d'une base de données spécifiée par un jeu de conditions.

*Type de retour* : FLOAT

##### Syntaxe

DVAR(Base\_de\_données; "en-tête"; Conditions)

##### Paramètres

*Commentaire* : Plage marquant la base de données, *Type* : Une plage de chaînes

*Commentaire* : Chaîne marquant la colonne dans la base de données, *Type* : Texte

*Commentaire* : Plage marquant les conditions, *Type* : Une plage de chaînes

##### Exemples

DVAR(A1 :C5; "Salaire"; A9 :A11)

##### Fonctions en relation

[DVARP](#)

#### 8.1.3.12 DVARP

renvoie l'estimation de la variance d'une population entière qui utilise toutes les valeurs numériques dans une colonne d'une base de données spécifiée par un jeu de conditions.

*Type de retour* : FLOAT

##### Syntaxe

DVARP(Base\_de\_données; "en-tête"; Conditions)

##### Paramètres

*Commentaire* : Plage marquant la base de données, *Type* : Une plage de chaînes

*Commentaire* : Chaîne marquant la colonne dans la base de données, *Type* : Texte

*Commentaire* : Plage marquant les conditions, *Type* : Une plage de chaînes

##### Exemples

DVARP(A1 :C5; "Salaire"; A9 :A11)

##### Fonctions en relation

[DVAR](#)

#### 8.1.3.13 GETPIVOTDATA

Cherche les données de sommaire dans une table pivot.

*Type de retour* : FLOAT

##### Syntaxe

GETPIVOTDATA(Base de données; "Ventes")

##### Paramètres

*Commentaire* : Plage marquant la table-pivot, *Type* : Une plage de chaînes

*Commentaire* : Nom du champ dont vous voulez les données de sommaire, *Type* : Texte

## 8.1.4 Date et heure

### 8.1.4.1 CURRENTDATE

La fonction CURRENTDATE() renvoie la date actuelle. Elle est équivalente à la fonction TODAY.

*Type de retour* : Date

#### Syntaxe

CURRENTDATE()

#### Paramètres

#### Exemples

CURRENTDATE() renvoie "Samedi 13 avril 2002"

#### Fonctions en relation

[CURRENTTIME](#)

[TODAY](#)

### 8.1.4.2 CURRENTDATETIME

La fonction CURRENTDATETIME() renvoie la date et heure courante.

*Type de retour* : Date

#### Syntaxe

CURRENTDATETIME()

#### Paramètres

#### Exemples

CURRENTDATETIME() renvoie "Samedi 15 juillet 2000 19 :12 :01"

### 8.1.4.3 CURRENTTIME

La fonction CURRENTTIME() renvoie l'heure actuelle formatée avec les paramètres locaux.

*Type de retour* : Date

#### Syntaxe

CURRENTTIME()

#### Paramètres

#### Exemples

CURRENTTIME() renvoie "19 :12 :01"

### 8.1.4.4 DATE

La fonction DATE() renvoie la date formatée avec les paramètres locaux.

*Type de retour* : Texte

#### Syntaxe

DATE(année ;mois ;jour)

#### Paramètres

*Commentaire* : Année, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Mois, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : jour, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

DATE(2000 ;5 ;5) renvoie Vendredi 5 Mai 2000



#### 8.1.4.5 DATE2UNIX

La fonction DATE2UNIX() convertit une date et une valeur de temps en temps unix.

Un temps unix est le nombre de secondes après le 1er Janvier 1970 minuit.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

DATE2UNIX(date)

##### Paramètres

*Commentaire* : Date, *Type* : Texte

##### Exemples

DATE2UNIX("01/01/2000") renvoie 946,681,200

#### 8.1.4.6 DATEDIF

La fonction DATEDIF() renvoie la différence entre deux dates.

l'intervalle doit être un des suivants : « m » : mois ; « j » : jours ; « a » : années complètes ; « am » : mois excluant les années ; « aj » : jours excluant les années ; « mj » : jours excluant les mois et les années

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

DATEDIF(première date ; seconde date ; intervalle)

##### Paramètres

*Commentaire* : Première date, *Type* : Texte

*Commentaire* : Deuxième date, *Type* : Texte

*Commentaire* : intervalle, *Type* : Texte

##### Exemples

DATEDIF(A1;A2;"d") si A1 est "1er Janvier 1995" et si A2 est "15 Juin 1999" calcule le nombre de jours entre les deux dates soit 1626

##### Exemples

DATEDIF(A1;A2;"m") si A1 est "1er Janvier 1995" et si A2 est "15 Juin 1999" calcule le nombre de semaines entre les deux dates soit 53

#### 8.1.4.7 DATEVALUE

La fonction DATEVALUE renvoie un nombre représentant le jour, c'est-à-dire le nombre de jours écoulés depuis le 31 décembre 1899.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

DATEVALUE(date)

##### Paramètres

*Commentaire* : Date, *Type* : Texte

##### Exemples

DATEVALUE("2/22/2002") renvoie 37309

##### Fonctions en relation

[TIMEVALUE](#)

#### 8.1.4.8 DAY

La fonction DAY renvoie le jour d'une date. Si aucun paramètre n'est spécifié, le jour courant est retourné.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

DAY(date)

##### Paramètres

*Commentaire* : Date, *Type* : Texte

##### Exemples

DAY("2/22/2002") renvoie 22

##### Exemples

DAY(2323,1285) renvoie 11

##### Fonctions en relation

[MONTH](#)

[YEAR](#)

#### 8.1.4.9 DAYNAME

La fonction DAYNAME() renvoie le nom du jour de la semaine (1...7). Dans certains pays, le premier jour de la semaine est lundi, alors que dans d'autres, c'est dimanche.

*Type de retour* : Texte

##### Syntaxe

DAYNAME(jour de la semaine)

##### Paramètres

*Commentaire* : Numéro du jour dans la semaine (1...7), *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

DAYNAME(1) renvoie lundi (si la semaine commence le lundi)

##### Fonctions en relation

[WEEKDAY](#)

#### 8.1.4.10 DAYOFYEAR

La fonction DAYOFYEAR() renvoie le numéro du jour dans l'année (1...365).

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

DAYOFYEAR(année;mois;jour)

##### Paramètres

*Commentaire* : Année, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Mois, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : jour, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

DAYOFYEAR(2000;12;1) renvoie 336

##### Exemples

DAYOFYEAR(2000;2;29) renvoie 60

#### 8.1.4.11 DAYS

La fonction DAYS() renvoie la différence entre deux dates en jours.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

DAYS(date2; date1)

##### Paramètres

*Commentaire* : Première valeur de date (plus tôt), *Type* : Texte

*Commentaire* : Deuxième valeur de date, *Type* : Texte

##### Exemples

DAYS("22/02/2002"; "26/02/2002") renvoie 4

#### 8.1.4.12 DAYS360

La fonction DAYS360() renvoie le nombre de jours de "date1" à "date2" en utilisant un calendrier de 360 jours dans lequel tous les mois sont supposés avoir 30 jours. Si "méthode" est Faux (par défaut), la méthode US sera utilisée, sinon ça sera la méthode européenne.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

DAYS360(date1; date2; méthode)

##### Paramètres

*Commentaire* : Date1, *Type* : Texte

*Commentaire* : Date2, *Type* : Texte

*Commentaire* : Méthode, *Type* : Une valeur de vérité (VRAI ou FAUX)

##### Exemples

DAYS360("2/22/2002", "4/21/2002", FAUX) renvoie 59.

##### Fonctions en relation

[DAYS](#)

[MONTHS](#)

[WEEKS](#)

[YEARS](#)

#### 8.1.4.13 DAYSINMONTH

La fonction DAYSINMONTH() renvoie le nombre de jours dans le mois et l'année donnés.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

DAYSINMONTH(année;mois)

##### Paramètres

*Commentaire* : Année, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Mois, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

DAYSINMONTH(2000;2) renvoie 29

#### 8.1.4.14 DAYSINYEAR

La fonction DAYSINYEAR() renvoie le nombre de jours dans l'année donnée.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

DAYSINYEAR(année)

##### Paramètres

*Commentaire* : Année, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

DAYSINYEAR(2000) renvoie 366

#### 8.1.4.15 EASTERSUNDAY

La fonction EASTERSUNDAY() renvoie la date qui correspond au dimanche de Pâques dans l'année donnée en paramètre.

*Type de retour* : Date

##### Syntaxe

EASTERSUNDAY(année)

##### Paramètres

*Commentaire* : Année, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

EASTERSUNDAY(2003) renvoie "20 avril 2003"

#### 8.1.4.16 EDATE

La fonction EDATE renvoie la date qui est spécifiée par une date donnée et un nombre de mois avant ou après cette date.

*Type de retour* : Date

##### Syntaxe

EDATE(date;mois)

##### Paramètres

*Commentaire* : Date, *Type* : Texte

*Commentaire* : Mois, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

EDATE("2/22/2002"; 3) renvoie "5/22/2002"

##### Exemples

EDATE("3/31/2002"; -1) renvoie "2/28/2002"

##### Fonctions en relation

[DATE](#)

[EOMONTH](#)

#### 8.1.4.17 EOMONTH

La fonction EOMONTH renvoie le dernier jour du mois spécifié par une date et le nombre de mois depuis cette date.

*Type de retour* : Date

##### Syntaxe

EOMONTH(date; mois)

##### Paramètres

*Commentaire* : Date, *Type* : Texte

*Commentaire* : Mois, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

EOMONTH("2/22/2002"; 3) renvoie "5/31/2002"

##### Exemples

EOMONTH("3/12/2002"; -1) renvoie "2/28/2002"

##### Exemples

EOMONTH("3/12/2002"; 0) renvoie "3/31/2002"

##### Fonctions en relation

[EDATE](#)

[MONTH](#)

#### 8.1.4.18 HOUR

La fonction HOUR renvoie l'heure d'un temps. Si aucun paramètre n'est spécifié, l'heure courante est retournée.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

HOUR(temps)

##### Paramètres

*Commentaire* : Heure, *Type* : Texte

##### Exemples

HOUR("22 :10 :12") renvoie 22

##### Exemples

HOUR(0,1285) renvoie 3

##### Fonctions en relation

[MINUTE](#)

[SECOND](#)

#### 8.1.4.19 HOURS

La fonction HOURS() renvoie la valeur des heures dans une expression de temps.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

HOURS(temps)

##### Paramètres

*Commentaire* : Heure, *Type* : Texte

##### Exemples

HOURS("10 :5 :2") renvoie 10

#### 8.1.4.20 ISLEAPYEAR

La fonction ISLEAPYEAR() renvoie Vrai si l'année donnée est bissextile.

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISLEAPYEAR(année)

##### Paramètres

*Commentaire* : Année, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

ISLEAPYEAR(2000) renvoie Vrai

#### 8.1.4.21 ISOWEEKNUM

La fonction ISOWEEKNUM() renvoie le numéro de semaine qui contient la date. Notez que cette fonction se plie au standard ISO8601 : une semaine standard commence toujours un lundi et finit un dimanche. La première semaine de l'année est celle qui contient le premier jeudi de l'année.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

ISOWEEKNUM(date)

##### Paramètres

*Commentaire* : Date, *Type* : Texte

##### Exemples

ISOWEEKNUM(A1) renvoie 51 quand A1 vaut "21 déc".

##### Fonctions en relation

[WEEKNUM](#)

#### 8.1.4.22 MINUTE

La fonction MINUTE renvoie les minutes d'un temps. Si aucun paramètre n'est spécifié, la minute courante est retournée.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

MINUTE(temps)

##### Paramètres

*Commentaire* : Heure, *Type* : Texte

##### Exemples

MINUTE("22 :10 :12") renvoie 10

##### Exemples

MINUTE(0,1234) renvoie 57

##### Fonctions en relation

[HOUR](#)  
[SECOND](#)

#### 8.1.4.23 MINUTES

La fonction MINUTES() renvoie la valeur des minutes dans une expression de temps.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

MINUTES(temps)

##### Paramètres

*Commentaire* : Heure, *Type* : Texte

##### Exemples

MINUTES("10 :5 :2") renvoie 5

#### 8.1.4.24 MONTH

La fonction MONTH renvoie le mois d'une date. Si aucun paramètre n'est spécifié, le mois courant est retourné.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

MONTH(date)

##### Paramètres

*Commentaire* : Date, *Type* : Texte

##### Exemples

MONTH("2/22/2002") renvoie 2

##### Exemples

MONTH(2323,1285) renvoie 5

##### Fonctions en relation

DAY  
YEAR

#### 8.1.4.25 MONTHNAME

La fonction MONTHNAME() renvoie le nom du mois (1...12).

*Type de retour* : Texte

##### Syntaxe

MONTHNAME(Nombre)

##### Paramètres

*Commentaire* : Numéro du mois (1..12), *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

MONTHNAME(5) renvoie mai

#### 8.1.4.26 MONTHS

La fonction MONTHS() renvoie la différence entre deux dates en mois. Le troisième paramètre indique le mode de calcul : si le mode est 0, MONTHS() renvoie le nombre maximal possible de mois entre ces jours. Si le mode est 1, elle renvoie le nombre de mois complets entre elles.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

### Syntaxe

MONTHS(date2; date1 ; mode)

### Paramètres

*Commentaire* : Première valeur de date (plus tôt), *Type* : Texte

*Commentaire* : Deuxième valeur de date, *Type* : Texte

*Commentaire* : Mode de calcul, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

MONTHS("18/1/2002"; "26/2/2002"; 0) renvoie 1, car il y a 1 mois et 8 jours entre les dates

### Exemples

MONTHS("19/1/2002", "26/2/2002", 1) renvoie 0, car il n'y a pas un mois entier de séparation, en commençant par le premier jour du mois

#### 8.1.4.27 NETWORKDAY

La fonction NETWORKDAY() renvoie le nombre de jours ouvrés entre la date de début et celle de fin.

Les vacances doivent avoir une des valeurs suivantes : nombre = jours à ajouter, une date seule ou un intervalle de dates.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

### Syntaxe

NETWORKDAY (date de début; date de fin ; vacances)

### Paramètres

*Commentaire* : Date de début, *Type* : Texte

*Commentaire* : Date de fin, *Type* : Texte

*Commentaire* : Vacances , *Type* : Texte

### Exemples

NETWORKDAY("01/01/2001"; "01/08/2001") renvoie 5 jours ouvrés

### Exemples

NETWORKDAY("01/01/2001"; "01/08/2001"; 2) renvoie 3 jours ouvrés

#### 8.1.4.28 NOW

La fonction NOW() renvoie la date et l'heure courante. Elle est identique à CURRENTDATETIME et fournie pour compatibilité avec d'autres applications.

*Type de retour* : Date

### Syntaxe

NOW()

### Paramètres

### Exemples

NOW() renvoie "Samedi 13 avril 2002 19 :12 :01"

### Fonctions en relation

CURRENTTIME

TODAY



#### 8.1.4.29 SECOND

La fonction SECOND renvoie les secondes d'un temps. Si aucun paramètre n'est spécifié, la seconde courante est retournée.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

SECOND(temps)

##### Paramètres

*Commentaire* : Heure, *Type* : Texte

##### Exemples

SECOND("22 :10 :12") renvoie 12

##### Exemples

SECOND(0,1234) renvoie 42

##### Fonctions en relation

[HOUR](#)

[MINUTE](#)

#### 8.1.4.30 SECONDS

La fonction SECONDS() renvoie la valeur des secondes dans une expression de temps.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

SECONDS(temps)

##### Paramètres

*Commentaire* : Heure, *Type* : Texte

##### Exemples

SECONDS("10 :5 :2") renvoie 2

#### 8.1.4.31 TIME

La fonction TIME() renvoie l'heure formatée avec les paramètres locaux.

*Type de retour* : Texte

##### Syntaxe

TIME(heures ;minutes ;secondes)

##### Paramètres

*Commentaire* : Heures, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Minutes, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Secondes, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

TIME(10 ;2 ;2) renvoie 10 :02 :02

##### Exemples

TIME(10 ;70 ;0) renvoie 11 :10 :0

##### Exemples

TIME(10 ;-40 ;0) renvoie 9 :20 :0

#### 8.1.4.32 TIMEVALUE

La fonction TIMEVALUE() renvoie un nombre (entre 0 et 1) représentant l'heure du jour.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

TIMEVALUE(temps)

##### Paramètres

*Commentaire* : Heure, *Type* : Texte

##### Exemples

TIMEVALUE("10 :05 :02") renvoie 0,42

##### Fonctions en relation

[DATEVALUE](#)

#### 8.1.4.33 TODAY

La fonction TODAY() renvoie la date actuelle.

*Type de retour* : Date

##### Syntaxe

TODAY()

##### Paramètres

##### Exemples

TODAY() renvoie "Samedi 13 avril 2002"

##### Fonctions en relation

[CURRENTTIME](#)

[NOW](#)

#### 8.1.4.34 UNIX2DATE

La fonction UNIX2DATE() convertit un temps unix en une date et une valeur de temps.

Un temps unix est le nombre de secondes après le 1er Janvier 1970 minuit.

*Type de retour* : Date

##### Syntaxe

UNIX2DATE(unixtime)

##### Paramètres

*Commentaire* : Temps unix, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

UNIX2DATE(0) renvoie 1970-01-01

#### 8.1.4.35 WEEKDAY

La fonction WEEKDAY() renvoie le jour de la semaine d'une date donnée. Si la méthode est 1 (par défaut), WEEKDAY() renvoie 1 pour dimanche, 2 pour lundi, etc. Si la méthode est 2, lundi est 1, mardi est 2, etc. et si la méthode est 3, WEEKDAY() renvoie 0 pour lundi, 1 pour mardi, etc.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

WEEKDAY(date ; méthode)

### Paramètres

*Commentaire* : Date, *Type* : Texte

*Commentaire* : Méthode (optionnel), *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

WEEKDAY("2002-02-22" ; 2) renvoie 5

### Fonctions en relation

[DAYNAME](#)

#### 8.1.4.36 WEEKNUM

La fonction WEEKNUM() renvoie le numéro non-ISO de semaine à laquelle la date correspond ;

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

### Syntaxe

WEEKNUM(date ; méthode)

### Paramètres

*Commentaire* : Date, *Type* : Texte

*Commentaire* : Méthode (optionnel), *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

NO.SEMAINE(A1 ; 1) renvoie 11 quand A1 est "9 Mars 2008". Numéro de la semaine dans l'année avec une semaine commençant le Dimanche (11)

### Exemples

WEEKNUM(A1 ; 2) renvoie 10 when A1 is "9th of March 2008". Number of the week in the year, with a week beginning on Monday (10)

### Fonctions en relation

[ISOWEEKNUM](#)

#### 8.1.4.37 WEEKS

La fonction WEEKS() renvoie l'écart entre deux dates en semaines. Le troisième paramètre indique le mode de calcul : si le mode est 0, WEEKS() renvoie le nombre maximal possible de semaines entre ces jours. Si le mode est 1, elle ne renvoie que le nombre de semaines entières entre elles.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

### Syntaxe

WEEKS(date2 ; date1 ; mode)

### Paramètres

*Commentaire* : Première valeur de date (plus tôt), *Type* : Texte

*Commentaire* : Deuxième valeur de date, *Type* : Texte

*Commentaire* : Mode de calcul, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

WEEKS("18/2/2002" ; "26/2/2002" ; 0) renvoie 1, parce qu'il y a une semaine et un jour entre

### Exemples

WEEKS("19/02/2002" ; "19/02/2002" ; 1) renvoie 0, car il n'y a pas une semaine entière entre, en commençant par le premier jour de la semaine (lundi ou dimanche en fonction de vos réglages locaux)

#### 8.1.4.38 WEEKSINYEAR

La fonction WEEKSINYEAR() renvoie le nombre de semaines dans l'année donnée.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

WEEKSINYEAR(année)

##### Paramètres

*Commentaire* : Année, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

WEEKSINYEAR(2000) renvoie 52

#### 8.1.4.39 WORKDAY

La fonction WORKDAY() renvoie la date des jours ouvrés depuis la date de début..

Les vacances doivent avoir une des valeurs suivantes : nombre = jours à ajouter, une date seule ou un intervalle de dates.

*Type de retour* : Date

##### Syntaxe

WORKDAY (date début ; jours ; vacances)

##### Paramètres

*Commentaire* : Date de début, *Type* : Texte

*Commentaire* : Jours ouvrés, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Vacances , *Type* : Texte

##### Exemples

WORKDAY(B9;2;D3 :E4) B9 est "01/01/2001", D3 est "01/03/2001", D4 est "01/04/2003"  
jours ouvrés "Vendredi 5 janvier 2001"

#### 8.1.4.40 YEAR

La fonction YEAR renvoie l'année d'une date. Si aucun paramètre n'est spécifié, l'année courante est retournée.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

YEAR(date)

##### Paramètres

*Commentaire* : Date, *Type* : Texte

##### Exemples

YEAR("2/22/2002") renvoie 2002

##### Exemples

YEAR(2323,1285) renvoie 1906

##### Fonctions en relation

DAY

MONTH

#### 8.1.4.41 YEARFRAC

La fonction YEARFRAC() renvoie le nombre de jours complets entre la date de début et la date de fin en fonction de la base.

Base doit être une des valeurs suivantes : 0 = 30/360 US, 1 = Actual/actual, 2 = Actual/360, 3 = Actual/365, 4 = European 30/360

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

FRACTION.ANNEE (date début ; date fin ; base)

##### Paramètres

*Commentaire* : Première date, *Type* : Texte

*Commentaire* : Deuxième date, *Type* : Texte

*Commentaire* : intervalle, *Type* : Texte

#### 8.1.4.42 YEARS

La fonction YEARS() renvoie la différence entre deux dates en années. Le troisième paramètre indique le mode de calcul : si le mode est 0, YEARS() renvoie le nombre maximal possible d'années entre ces jours. Si le mode est 1, elle ne renvoie que les années entières, en commençant par le 1er janvier et en terminant par le 31 décembre.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

YEARS(date2 ; date1 ; mode)

##### Paramètres

*Commentaire* : Première valeur de date (plus tôt), *Type* : Texte

*Commentaire* : Deuxième valeur de date, *Type* : Texte

*Commentaire* : Mode de calcul, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

YEARS("19/02/2001" ; "26/02/2002" ; 0) renvoie 1, car il y a un an et 7 jours entre les dates

##### Exemples

YEARS("19/02/2002" ; "26/02/2002" ; 1) renvoie 0, car il n'y a pas une année complète entre les dates lorsqu'on commence le premier jour de l'année

### 8.1.5 Technique

#### 8.1.5.1 BASE

La fonction BASE() convertit un nombre d'une base 10 en une valeur de chaîne dans une base cible entre 2 et 36.

*Type de retour* : Texte

##### Syntaxe

BASE(nombre ; base ; prec)

##### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : LongueurMini, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

BASE(128;8) renvoie "200"

### 8.1.5.2 BESSELI

La fonction BESSELI() renvoie la fonction Bessel  $I_n(x)$  modifiée.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

BESSELI(X;N)

#### Paramètres

*Commentaire* : Où la fonction est évaluée, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Ordre de la fonction, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

BESSELI(0,7;3) renvoie 0,007367374

#### Fonctions en relation

[BESSELJ](#)

[BESSELK](#)

[BESSELY](#)

### 8.1.5.3 BESSELJ

La fonction BESSELJ() renvoie la fonction BESSEL.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

BESSELJ(X;N)

#### Paramètres

*Commentaire* : Où la fonction est évaluée, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Ordre de la fonction, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

BESSELJ(0,89;3) renvoie 0,013974004

#### Fonctions en relation

[BESSELI](#)

[BESSELK](#)

[BESSELY](#)

### 8.1.5.4 BESSELK

La fonction BESSELK() renvoie la fonction Bessel modifiée, qui est équivalente à la fonction Bessel évaluée pour des arguments purement imaginaires.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

BESSELK(X;N)

#### Paramètres

*Commentaire* : Où la fonction est évaluée, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Ordre de la fonction, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

BESSELK(3;9) renvoie 397,95880

#### Fonctions en relation

[BESSELI](#)

[BESSELJ](#)

[BESSELY](#)

#### 8.1.5.5 BESSELY

La fonction BESSELY() renvoie la fonction Bessel, qui est aussi nommée la fonction de Weber ou la fonction de Neumann.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

BESSELY(X;N)

##### Paramètres

*Commentaire* : Où la fonction est évaluée, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Ordre de la fonction, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

BESSELY(4;2) vaut 0,215903595

##### Fonctions en relation

BESSELI

BESSELJ

BESSELK

#### 8.1.5.6 BIN2DEC

La fonction BIN2DEC() renvoie la valeur formatée comme un nombre décimal.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

BIN2DEC(valeur)

##### Paramètres

*Commentaire* : La valeur à convertir, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

BIN2DEC("1010") renvoie 10

##### Exemples

BIN2DEC("11111") renvoie 31

#### 8.1.5.7 BIN2HEX

La fonction BIN2HEX() renvoie la valeur formatée comme un nombre hexadécimal.

*Type de retour* : Texte

##### Syntaxe

BIN2HEX(valeur)

##### Paramètres

*Commentaire* : La valeur à convertir, *Type* : Texte

*Commentaire* : La longueur minimum de la sortie, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

BIN2HEX("1010") renvoie "a"

##### Exemples

BIN2HEX("11111") "1f"

#### 8.1.5.8 BIN2OCT

La fonction BIN2OCT() renvoie la valeur formatée comme un nombre octal.

*Type de retour* : Texte

##### Syntaxe

BIN2OCT(valeur)

##### Paramètres

*Commentaire* : La valeur à convertir, *Type* : Texte

*Commentaire* : La longueur minimum de la sortie, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

BIN2OCT("1010") renvoie "12"

##### Exemples

BIN2OCT("11111") renvoie "37"

#### 8.1.5.9 COMPLEX

COMPLEX(réel,imag) renvoie un nombre complexe de la forme x+yi.

*Type de retour* : Texte

##### Syntaxe

COMPLEX(réel;imag)

##### Paramètres

*Commentaire* : Coefficient réel, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Coefficient imaginaire, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

COMPLEX(1,2;3,4) renvoie "1,2+3,4i"

##### Exemples

COMPLEX(0;-1) renvoie "-i"

#### 8.1.5.10 CONVERT

La fonction CONVERT() renvoie une conversion d'un système de mesure à un autre.

Unités de masse gérées : g (gramme), sg (pieces), lbm (pound=livre), u (masse atomique), ozm (ounce), stone, ton, grain, pweight (pennyweight), hweight (hundredweight), bron (gross registered ton).

Unités de distance gérées : m (mètre), in (inch=pouce), ft (feet=pied), mi (mile), Nmi (mile nautique), ang (Angstrom), parse, lightyear (année lumière).

Unités de pression gérées : Pa (Pascal), atm (atmosphère), mmHg (mm de mercure), psi, Torr.

Unités de force gérées : N (Newton), dyn, pound.

Unités d'énergie gérées : J (Joule), e (erg), c (calorie thermodynamique), cal (IT calorie), eV (électron-volt), HPh (Cheval-vapeur.heure), Wh (Watt.heure), flb (foot-pound), BTU.

Unités de puissance gérées : W (Watt), HP (horsepower=cheval-vapeur), PS (Pferdestaerke=cheval-vapeur).

Unités de magnétisme gérées : T (Tesla), ga (Gauss).

Unités de température gérées : C (Celsius), F (Fahrenheit), K (Kelvin).



Unités de volume gérées : l (litre), tsp (cuiller à café), tbs (cuiller à soupe), oz (ounce liquid), cup, pt (pint), qt (quart), gal (gallon), barrel, m3 (mètre cube), mi3 (mile cubique), Nmi3 (mile nautique cubique), in3 (pouce cube), ft3 (pied cube), yd3 (yard cube).

Unités de surface gérées : m2 (mètre carré), mi2 (mile carré), Nmi2 (mile nautique carré), in2 (pouce carré), ft2 (pied carré), yd2 (yard carré), acre, ha (hectare).

Unités de vitesse gérées : m/s (mètres par seconde), m/h (mètres par heure), mph (miles par heure), kn (nœud).

Pour les unités métriques, n'importe lequel des préfixes suivants peut être utilisé : E (exa, 1E+18), P (peta, 1E+15), T (tera, 1E+12), G (giga, 1E+09), M (mega, 1E+06), k (kilo, 1E+03), h (hecto, 1E+02), e (deca, 1E+01), d (deci, 1E-01), c (centi, 1E-02), m (milli, 1E-03), u (micro, 1E-06), n (nano, 1E-09), p (pico, 1E-12), f (femto, 1E-15), a (atto, 1E-18).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

CONVERT(Nombre;de l'unité;vers l'unité)

### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Unité source, *Type* : Texte

*Commentaire* : Unité finale, *Type* : Texte

### Exemples

CONVERT(32;"C";"F") vaut 89,6

### Exemples

CONVERT(3;"lbm";"kg") vaut 1,3608

### Exemples

CONVERT(7,9;"cal";"J") vaut 33,0757

## 8.1.5.11 DEC2BIN

La fonction DEC2BIN() renvoie la valeur formatée comme nombre binaire.

*Type de retour* : Texte

### Syntaxe

DEC2BIN(valeur)

### Paramètres

*Commentaire* : La valeur à convertir, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : La longueur minimum de la sortie, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

DEC2BIN(12) renvoie "1100"

### Exemples

DEC2BIN(55) renvoie "110111"

## 8.1.5.12 DEC2HEX

La fonction DEC2HEX() renvoie la valeur formatée comme un nombre hexadécimal.

*Type de retour* : Texte

### Syntaxe

DEC2HEX(valeur)

#### Paramètres

*Commentaire* : La valeur à convertir, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : La longueur minimum de la sortie, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

DEC2HEX(12) renvoie "c"

#### Exemples

DEC2HEX(55) renvoie "37"

#### 8.1.5.13 DEC2OCT

La fonction DEC2OCT() renvoie la valeur formatée comme un nombre octal.

*Type de retour* : Texte

#### Syntaxe

DEC2OCT(valeur)

#### Paramètres

*Commentaire* : La valeur à convertir, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : La longueur minimum de la sortie, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

DEC2OCT(12) renvoie "14"

#### Exemples

DEC2OCT(55) renvoie "67"

#### 8.1.5.14 DELTA

La fonction DELTA() renvoie 1 si x vaut y, sinon elle renvoie 0. La valeur par défaut de y est 0.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

DELTA(x; y)

#### Paramètres

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

DELTA(1,2;3,4) renvoie 0

#### Exemples

DELTA(3;3) renvoie 1

#### Exemples

DELTA(1;VRAI) renvoie 1

#### 8.1.5.15 ERF

La fonction ERF() renvoie la fonction erreur. Avec un seul argument, ERF() renvoie la fonction erreur entre 0 et cet argument.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

ERF(limite basse;limite haute)

### Paramètres

*Commentaire* : Délimiteur bas, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Délimiteur haut, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

ERF(0,4) vaut 0,42839236

### Fonctions en relation

[ERFC](#)

#### 8.1.5.16 ERFC

La fonction ERFC() renvoie la fonction d'erreur complémentaire.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

ERFC(Limite basse;limite haute)

### Paramètres

*Commentaire* : Délimiteur bas, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Délimiteur haut, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

ERFC(0,4) vaut 0,57160764

### Fonctions en relation

[ERF](#)

#### 8.1.5.17 GESTEP

La fonction GESTEP() renvoie 1 si x est supérieur ou égal à y, sinon elle renvoie 0. La valeur de y par défaut est 0.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

GESTEP(x; y)

### Paramètres

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

GESTEP(1,2;3,4) renvoie 0

### Exemples

GESTEP(3;3) renvoie 1

### Exemples

GESTEP(0,4;VRAI) renvoie 0

### Exemples

GESTEP(4;3) renvoie 1

#### 8.1.5.18 HEX2BIN

La fonction HEX2BIN() renvoie la valeur formatée comme nombre binaire.

*Type de retour* : Texte

##### Syntaxe

HEX2BIN(valeur)

##### Paramètres

*Commentaire* : La valeur à convertir, *Type* : Texte

##### Exemples

HEX2BIN("A") renvoie "1010"

##### Exemples

HEX2BIN("37") renvoie "110111"

#### 8.1.5.19 HEX2DEC

La fonction HEX2DEC() renvoie la valeur formatée comme un nombre décimal.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

HEX2DEC(valeur)

##### Paramètres

*Commentaire* : La valeur à convertir, *Type* : Texte

##### Exemples

HEX2DEC("a") renvoie 10

##### Exemples

HEX2DEC("37") renvoie 55

#### 8.1.5.20 HEX2OCT

La fonction HEX2OCT() renvoie la valeur formatée comme un nombre octal.

*Type de retour* : Texte

##### Syntaxe

HEX2OCT(valeur)

##### Paramètres

*Commentaire* : La valeur à convertir, *Type* : Texte

##### Exemples

HEX2OCT("a") renvoie "12"

##### Exemples

HEX2OCT("37") renvoie "67"

#### 8.1.5.21 IMABS

IMABS(nombre complexe) renvoie la norme d'un nombre complexe de forme  $x+yi$ .

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

IMABS(nombre complexe)

**Paramètres**

*Commentaire* : Nombre complexe, *Type* : Texte

**Exemples**

IMABS("1,2+5i") renvoie 5,1419

**Exemples**

IMABS("-i") renvoie 1

**Exemples**

IMABS("12") renvoie 12

**8.1.5.22 IMAGINARY**

IMAGINARY(chaine) renvoie le coefficient imaginaire d'un nombre complexe.

*Type de retour* : Double

**Syntaxe**

IMAGINARY(chaine)

**Paramètres**

*Commentaire* : Nombre complexe, *Type* : Texte

**Exemples**

IMAGINARY("1,2+3,4i") renvoie 3,4

**Exemples**

IMAGINARY("1,2") renvoie 0

**8.1.5.23 IMARGUMENT**

IMARGUMENT(nombre complexe) renvoie l'argument d'un nombre complexe de forme  $x+yi$ .

*Type de retour* : Texte

**Syntaxe**

IMARGUMENT(nombre complexe)

**Paramètres**

*Commentaire* : Nombre complexe, *Type* : Texte

**Exemples**

IMARGUMENT("1,2+5i") renvoie 0,6072

**Exemples**

IMARGUMENT("-i") renvoie -1.57079633

**Exemples**

IMARGUMENT("12") renvoie "#Div/0"

**8.1.5.24 IMCONJUGATE**

IMCONJUGATE(nombre complexe) renvoie le conjugué d'un nombre complexe de forme  $x+yi$ .

*Type de retour* : Texte

**Syntaxe**

IMCONJUGATE(nombre complexe)

**Paramètres**

*Commentaire* : Nombre complexe, *Type* : Texte

**Exemples**

IMCONJUGATE("1,2+5i") renvoie "1,2-5i"

**Exemples**

IMCONJUGATE("-i") renvoie "i"

**Exemples**

IMCONJUGATE("12") renvoie "12"

#### 8.1.5.25 IMCOS

IMCOS(chaine) renvoie le cosinus d'un nombre complexe.

*Type de retour* : Texte

##### Syntaxe

IMCOS(chaine)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

##### Exemples

IMCOS("1+i") renvoie "0,83373-0,988898i"

##### Exemples

IMCOS("12i") renvoie 81 377.4

#### 8.1.5.26 IMCOSH

IMCOSH(chaine) renvoie le cosinus hyperbolique d'un nombre complexe.

*Type de retour* : Texte

##### Syntaxe

IMCOSH(chaine)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

##### Exemples

IMCOSH("1+i") renvoie "0.83373+0.988898i"

##### Exemples

IMCOSH("12i") renvoie 0.84358

#### 8.1.5.27 IMCOT

IMCOT(chaine) renvoie la cotangente d'un nombre complexe.

*Type de retour* : Texte

##### Syntaxe

IMCOT(chaine)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

##### Exemples

IMCOT("1+i") renvoie « 0.21762-0.86801i »

#### 8.1.5.28 IMCSC

IMCSC(string) renvoie la cosécante d'un nombre complexe.

*Type de retour* : Texte

##### Syntaxe

IMCSC(chaine)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

##### Exemples

IMCSC("1+i") renvoie « 0.62151-0.30393i »

#### 8.1.5.29 IMCSCH

IMCSCH(chaine) renvoie la cosécante hyperbolique d'un nombre complexe.

*Type de retour* : Texte

##### Syntaxe

IMCSCH(chaine)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

##### Exemples

IMCSCH("1+i") renvoie « 0.30393-i0.62151 »

#### 8.1.5.30 IMDIV

IMDIV() renvoie la division de plusieurs nombres complexes de forme x+iy.

*Type de retour* : Texte

##### Syntaxe

IMDIV(valeur;valeur;...)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes

*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes

*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes

*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes

*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes

##### Exemples

IMDIV(1,2;"3,4+5i") renvoie "0,111597-0,164114i"

##### Exemples

IMDIV("12+i";"12-i") renvoie "0,986207+0,16551i"

#### 8.1.5.31 IMEXP

IMEXP(chaine) renvoie l'exponentielle d'un nombre complexe.

*Type de retour* : Texte

##### Syntaxe

IMEXP(chaine)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

##### Exemples

IMEXP("2-i") renvoie "3,99232-6,21768i"

##### Exemples

IMEXP("12i") renvoie « 0.843854-0.536573i »

#### 8.1.5.32 IMLN

IMLN(chaine) renvoie le logarithme naturel d'un nombre complexe.

*Type de retour* : Texte

##### Syntaxe

IMLN(chaine)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

##### Exemples

IMLN("3-i") renvoie "1,15129-0,321751i"

##### Exemples

IMLN("12") renvoie 2,48491

#### 8.1.5.33 IMLOG10

IMLLOG10(chaine) renvoie le logarithme en base 10 d'un nombre complexe.

*Type de retour* : Texte

##### Syntaxe

IMLOG10(chaine)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

##### Exemples

IMLOG10("3+4i") renvoie "0.69897+0.402719i"

#### 8.1.5.34 IMLOG2

IMLOG2(chaine) renvoie le logarithme en base-2 d'un nombre complexe.

*Type de retour* : Texte

##### Syntaxe

IMLOG2(chaine)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

##### Exemples

IMLOG2("3+4i") renvoie "2.321928+1.337804i"

#### 8.1.5.35 IMPOWER

IMPOWER(chaine) renvoie un nombre complexe élevé à une puissance.

*Type de retour* : Texte

##### Syntaxe

IMPOWER(chaine)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

*Commentaire* : Puissance, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

IMPOWER("4-i";2) renvoie "15-8i"

##### Exemples

IMPOWER("1.2";2) renvoie 1.44



### 8.1.5.36 IMPRODUCT

IMPRODUCT() renvoie le produit de plusieurs nombres complexes de forme  $x+yi$ .

*Type de retour* : Texte

#### Syntaxe

IMPRODUCT(valeur;valeur;...)

#### Paramètres

*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes

*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes

*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes

*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes

*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes

#### Exemples

IMPRODUCT(1,2;"3,4+5i") renvoie "4,08+6i"

#### Exemples

IMPRODUCT(1,2;"1i") renvoie "+1,2i"

### 8.1.5.37 IMREAL

IMREAL(chaine) renvoie le coefficient réel d'un nombre complexe.

*Type de retour* : Double

#### Syntaxe

IMREAL(chaine)

#### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

#### Exemples

IMREAL("1,2+3,4i") renvoie 1,2

#### Exemples

IMREAL("1.2i") renvoie 0

### 8.1.5.38 IMSEC

IMSEC(chaine) renvoie la sécante d'un nombre complexe.

*Type de retour* : Texte

#### Syntaxe

IMSEC(chaine)

#### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

#### Exemples

IMSEC("1+i") renvoie « 0.49833+i0.59108 »

### 8.1.5.39 IMSECH

IMSECH(chaine) renvoie la sécante hyperbolique d'un nombre complexe.

*Type de retour* : Texte

#### Syntaxe

IMSECH(chaine)

#### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

#### Exemples

IMSECH("1+i") renvoie « 0.49833-i0.59108 »

#### 8.1.5.40 IMSIN

La fonction IMSIN(chaine) renvoie le sinus d'un nombre complexe.

*Type de retour* : Texte

##### Syntaxe

IMSIN(chaine)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

##### Exemples

IMSIN("1+i") renvoie "1,29846+0,634964i"

##### Exemples

IMSIN("1,2") renvoie -0,536573

#### 8.1.5.41 IMSINH

La fonction IMSINH(chaine) renvoie le sinus hyperbolique d'un nombre complexe.

*Type de retour* : Texte

##### Syntaxe

IMSINH(chaine)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

##### Exemples

IMSINH("1+i") renvoie "0.63496+1.29846i"

##### Exemples

IMSINH("1.2") renvoie 1.50946

#### 8.1.5.42 IMSQRT

IMSQRT(chaine) renvoie la racine carrée d'un nombre complexe.

*Type de retour* : Texte

##### Syntaxe

IMSQRT(chaine)

##### Paramètres

*Commentaire* : Nombre complexe, *Type* : Texte

##### Exemples

IMSQRT("1+i") renvoie "1,09868+0,45509i"

##### Exemples

IMSQRT("1.2i") renvoie "0.774597+0.774597i"

#### 8.1.5.43 IMSUB

IMSUB() renvoie la différence entre plusieurs nombres complexes de forme  $x+yi$ .

*Type de retour* : Texte

##### Syntaxe

IMSUB(valeur;valeur;...)

**Paramètres**

*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes  
*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes  
*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes  
*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes  
*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes

**Exemples**

IMSUB(1,2;"3,4+5i") renvoie "-2,2-5i"

**Exemples**

IMSUB(1,2;"1i") renvoie "1,2-i"

**8.1.5.44 IMSUM**

IMSUM() renvoie la somme de plusieurs nombres complexes de forme  $x+yi$ .

*Type de retour* : Texte

**Syntaxe**

IMSUM(valeur,valeur;...)

**Paramètres**

*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes  
*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes  
*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes  
*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes  
*Commentaire* : Nombre complexe, *Type* : Une plage de chaînes

**Exemples**

IMSUM(1,2;"3,4+5i") renvoie "4,6+5i"

**Exemples**

IMSUM(1,2;"1i") renvoie "1,2+i"

**8.1.5.45 IMTAN**

La fonction IMTAN(chaîne) renvoie la tangente d'un nombre complexe.

*Type de retour* : Texte

**Syntaxe**

IMTAN(chaîne)

**Paramètres**

*Commentaire* : Nombre complexe, *Type* : Texte

**Exemples**

IMTAN("1+i") renvoie "0.27175+1.08392i"

**Exemples**

IMTAN("12") renvoie 2.57215

**8.1.5.46 IMTANH**

La fonction IMTANH(chaîne) renvoie la tangente hyperbolique d'un nombre complexe.

*Type de retour* : Texte

**Syntaxe**

IMTANH(chaîne)

**Paramètres**

*Commentaire* : Nombre complexe, *Type* : Texte

**Exemples**

IMTAN("1+i") renvoie "1.08392+0.27175i"

**Exemples**

IMTANH("1.2") renvoie 0.83365

**8.1.5.47 OCT2BIN**

La fonction OCT2BIN() renvoie la valeur formatée comme nombre binaire.

*Type de retour* : Texte

**Syntaxe**

OCT2BIN(valeur)

**Paramètres**

*Commentaire* : La valeur à convertir, *Type* : Texte

*Commentaire* : La longueur minimum de la sortie, *Type* : Un nombre entier (comme 1, 132, 2344)

**Exemples**

OCT2BIN("12") renvoie "1010"

**Exemples**

OCT2BIN("55") renvoie "101101"

**8.1.5.48 OCT2DEC**

La fonction OCT2DEC() renvoie la valeur formatée comme un nombre décimal.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

**Syntaxe**

OCT2DEC(valeur)

**Paramètres**

*Commentaire* : La valeur à convertir, *Type* : Texte

**Exemples**

OCT2DEC("12") renvoie 10

**Exemples**

OCT2DEC("55") renvoie 45

**8.1.5.49 OCT2HEX**

La fonction OCT2HEX() renvoie la valeur formatée comme un nombre hexadécimal.

*Type de retour* : Texte

**Syntaxe**

OCT2HEX(valeur)

**Paramètres**

*Commentaire* : La valeur à convertir, *Type* : Texte

*Commentaire* : La longueur minimum de la sortie, *Type* : Un nombre entier (comme 1, 132, 2344)

**Exemples**

OCT2HEX("12") renvoie "A"

**Exemples**

OCT2HEX("55") renvoie "2D"

## 8.1.6 Financières

### 8.1.6.1 ACCRINT

La fonction ACCRINT renvoie l'intérêt accru pour un titre qui paie des intérêts périodiques. Les fréquences autorisées sont 1 - annuel, 2 - semi-annuel ou 4 - trimestriel. "Base" est le type de comptage de jours que vous voulez utiliser : 0 : 30 / 360 US (par défaut), 1 : vrais jours, 2 : vrais jours / 360, 3 : vrais jours / 365 ou 4 : 30 / 365 européen.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

ACCRINT(terme;premier intérêt;règlement;taux;par;fréquence;base)

#### Paramètres

*Commentaire* : Date de sortie, *Type* : Date

*Commentaire* : Premier intérêt, *Type* : Date

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Taux annuel d'un titre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Par valeur, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Nombre de paiements par an, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base de comptage des jours, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

ACCRINT("28/2/2001"; "31/8/2001"; "1/5/2001"; 0,1;1000; 2; 0) renvoie 16 944

#### Fonctions en relation

[ACCRINTM](#)

### 8.1.6.2 ACCRINTM

La fonction ACCRINTM renvoie l'intérêt accru pour un titre qui paie des intérêts à la date d'échéance. "Base" est le type de comptage de jours que vous voulez utiliser : 0 : 30 / 360 US (par défaut), 1 : vrais jours, 2 : vrais jours / 360, 3 : vrais jours / 365 ou 4 : 30 / 365 européen.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

ACCRINTM(terme;règlement;taux;par;base)

#### Paramètres

*Commentaire* : Date de sortie, *Type* : Date

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Taux annuel d'un titre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Par valeur, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base de comptage des jours, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

ACCRINTM("28/2/2001"; "31/8/2001"; 0,1; 100) renvoie 5,0278

#### Fonctions en relation

[ACCRINT](#)

### 8.1.6.3 AMORDEGRC

La fonction The AMORDEGRC calcule la valeur d'amortissement dans le système comptable français en utilisant la dépréciation dégressive.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

AMORDEGRC( Coût; DateAchat; DatefinPremièrePériode; résidu; période; taux; base)

### Paramètres

*Commentaire* : Coût, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Pv, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Fv, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

AMORDEGRC( 1000; "2006-02-01"; "2006-12-31"; 10; 0; 0.1; 1 ) renvoie 228

### Fonctions en relation

[AMORLINC](#)

[DB](#)

[DDB](#)

[YEARFRAC](#)

#### 8.1.6.4 AMORLINC

La fonction AMORLINC calcule la valeur d'amortissement pour le système comptable français utilisant la dépréciation linéaire.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

AMORLINC( Coût; DateAchat; DatefinPremièrePériode; résidu; période; taux; base)

### Paramètres

*Commentaire* : P, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Pv, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Fv, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

AMORLINC( 1000; "2004-02-01"; "2004-12-31"; 10; 0; 0.1; 1 ) renvoie 91.256831

### Fonctions en relation

[AMORDEGRC](#)

[DB](#)

[DDB](#)

[YEARFRAC](#)

#### 8.1.6.5 COMPOUND

La fonction COMPOUND() renvoie la valeur d'un investissement, étant donnés le principal, le taux d'intérêt nominal, la fréquence de composition et le temps. Par exemple : 5 000 € à 12 % d'intérêt composé par trimestre pendant 5 ans deviendra COMPOUND(5000; 0,12; 4;5), soit 9 030,56 €.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

COMPOUND(initial; intérêt; périodes; périodes\_par\_année)

### Paramètres

*Commentaire* : Principal, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Taux d'intérêt, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Périodes par année, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Années, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

COMPOUND(5000; 0,12; 4; 5) vaut 9 030,56

#### 8.1.6.6 CONTINUOUS

La fonction CONTINUOUS() calcule le retour sur un intérêt continuellement composé, étant donné le principal, le taux d'intérêt, le taux nominal et le temps en années. Par exemple : 1 000 € gagnant 10 % pendant 1 an deviennent CONTINUOUS(1000; 0,1; 1), soit 1 105,17 €.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

CONTINUOUS(principal; intérêt; années)

##### Paramètres

*Commentaire* : Principal, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Taux d'intérêt, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Années, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

CONTINUOUS(1000; 0,1; 1) vaut 1 105,17

#### 8.1.6.7 COUPNUM

La fonction COUPNUM renvoie le nombre de coupons qui doivent être payés entre le début et le terme. "Base" est le type de jour, sachant que vous pouvez utiliser : US 30 / 360 (par défaut), 1 : vrais jours, 2 : vrais jours / 360, 3 : vrais jours / 365 ou 4 : Européen 30 / 365.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

COUPNUM(début, échéance, fréquence, base)

##### Paramètres

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Échéance, *Type* : Date

*Commentaire* : La fréquence, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base de comptage des jours, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

COUPNUM("28/2/2001"; "31/8/2001"; 2; 0) renvoie 1

#### 8.1.6.8 CUMIPMT

Calcule le montant des intérêts cumulatifs.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

CUMIPMT(rate, periods, value, start, end, type)

##### Paramètres

*Commentaire* : taux, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : périodes, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : valeur, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : start, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Fin, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : type, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

CUMIPMT( 0.06/12; 5\*12; 100000; 5; 12; 0 ) vaut -3562,187023

##### Fonctions en relation

[IPMT](#)

[CUMPRINC](#)

### 8.1.6.9 CUMPRINC

Calcule le principal paiement cumulatif.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

CUMPRINC(taux, périodes, valeur, début, fin, type)

#### Paramètres

*Commentaire* : taux, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : périodes, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : valeur, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : start, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Fin, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : type, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

CUMPRINC( 0.06/12; 5\*12; 100000; 5; 12; 0 ) vaut -11904,054201

#### Fonctions en relation

[PPMT](#)

[CUMIPMT](#)

### 8.1.6.10 DB

La fonction DB() calcule la dépréciation d'un bien pour une période donnée en utilisant la méthode de balance à déclin fixé. "Mois" est optionnel, si omis, il est supposé valoir 12.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

DB(coût; valeur résiduelle; vie; période [;mois])

#### Paramètres

*Commentaire* : Coût, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Résidu, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Vie, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Période, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Mois, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

DB(8000;400;6;3) vaut 1 158,40

#### Exemples

DB(8000;400;6;3;2) vaut 1 783,41

#### Fonctions en relation

[DDB](#)

[SLN](#)

### 8.1.6.11 DDB

La fonction DDB() calcule la dépréciation d'un bien pour une période donnée en utilisant la méthode de déclinaison arithmétique. Le facteur est optionnel, si omis, il est supposé valoir 2. Tous les paramètres doivent être supérieurs à 0.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

DDB(coût; valeur résiduelle; vie; période [;facteur])



### Paramètres

*Commentaire* : Coût, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)  
*Commentaire* : Résidu, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)  
*Commentaire* : Vie, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)  
*Commentaire* : Période, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)  
*Commentaire* : Facteur, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

DDB(75000;1;60;12;2) renvoie 1721,81

### Fonctions en relation

SLN

### 8.1.6.12 DISC

La fonction DISC renvoie le taux d'escompte pour une titre. "Base" est le type de jour selon que vous voulez utiliser : 0 : 30 / 360 US (par défaut), 1 : vrais jours, 2 : vrais jours / 360, 3 : vrais jours / 365 ou 4 : 30 / 365 européen.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

DISC(règlement; échéance; par; remise; base)

### Paramètres

*Commentaire* : Règlement, *Type* : Date  
*Commentaire* : Échéance, *Type* : Date  
*Commentaire* : Prix par valeur faciale de 100 €, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)  
*Commentaire* : Remise, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)  
*Commentaire* : Base de comptage des jours, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

DISC("28/2/2001"; "31/8/2001"; 12; 14) renvoie 0,2841

### Fonctions en relation

YEARFRAC

### 8.1.6.13 DOLLARDE

La fonction DOLLARDE() renvoie un prix en dollars exprimé en nombre décimal. "Dollar fractionnel" est le nombre à convertir et "fraction" est le dénominateur de la fraction.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

DOLLARDE(dollar fractionnel; fraction)

### Paramètres

*Commentaire* : Dollar fractionnel, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)  
*Commentaire* : Fraction, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

DOLLARDE(1,02; 16) - vaut pour 1 et 2/16 - renvoie 1,125

### Fonctions en relation

DOLLARFR  
TRUNC

#### 8.1.6.14 DOLLARFR

La fonction DOLLARFR() renvoie un prix en dollar exprimé comme une fraction. Le dollar décimal est le nombre à convertir et la fraction est le dénominateur de la fraction.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

DOLLARFR(dollar fractionnel; fraction)

##### Paramètres

*Commentaire* : Dollar décimal, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Fraction, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

DOLLARFR(1.125; 16) renvoie 1.02. (1 + 2/16)

##### Fonctions en relation

DOLLARDE  
TRUNC

#### 8.1.6.15 DURATION

Renvoie le nombre de périodes nécessaires pour un investissement pour obtenir une valeur désirée.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

DURATION(taux; pv; fv)

##### Paramètres

*Commentaire* : Vitesse, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur actuelle (PV), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur future (FV), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

DURATION(0,1; 1000; 2000) renvoie 7,27

##### Fonctions en relation

FV  
PV

#### 8.1.6.16 DURATION\_ADD

renvoie la durée Macauley en années de la sécurisation d'un intérêt fixe.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

DURATION\_ADD(règlement; début, coupon; échéance, rendement; fréquence; base)

##### Paramètres

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Échéance, *Type* : Date

*Commentaire* : Coupon, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Rendement, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : La fréquence, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

DURATION\_ADD( "1998-01-01" ; "2006-01-01" ; 0.08 ; 0.09 ; 2 ; 1 ) renvoie 5.9937749555

### Fonctions en relation

[MDURATION](#)

#### 8.1.6.17 EFFECT

La fonction EFFECT() calcule le rendement effectif pour un taux d'intérêt nominal (taux annuel ou APR). Par exemple : 8 % d'intérêts composés mensuellement fournissent un rendement effectif de EFFECT(0,08;12), soit 8,3 %.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

EFFECT(nominal;périodes)

### Paramètres

*Commentaire* : Taux nominal d'intérêt, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Périodes, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

EFFECT(0,08;12) vaut 0,083

### Fonctions en relation

[EFFECTIVE](#)

[NOMINAL](#)

#### 8.1.6.18 EFFECTIVE

La fonction EFFECTIVE() calcule le rendement effectif pour un taux d'intérêt nominal (taux annuel ou APR). Elle a le même effet que la fonction EFFECT.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

EFFECTIVE(nominal;périodes)

### Paramètres

*Commentaire* : Taux nominal d'intérêt, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Périodes, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Fonctions en relation

[EFFECT](#)

#### 8.1.6.19 EURO

La fonction EURO() convertit un Euro en monnaie nationale dans l'union monétaire européenne. La monnaie est l'une des suivantes : ATS (Autriche), BEF (Belgique), DEM (Allemagne), ESP (Espagne), FIM (Finlande), FRF (France), GRD (Grèce), IEP (Irlande), ITL (Italie), LUF (Luxembourg), NLG (Pays-Bas) ou PTE (Portugal).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

EURO(monnaie)

### Paramètres

*Commentaire* : Monnaie, *Type* : Texte

### Exemples

EURO("DEM") vaut 1,95583

### Fonctions en relation

[EUROCONVERT](#)

#### 8.1.6.20 EUROCONVERT

La fonction EURO() convertit un Euro en monnaie nationale dans l'union monétaire Européenne. La monnaie est l'une des suivantes : ATS (Autriche), BEF (Belgique), DEM (Allemagne), ESP (Espagne), FIM (Finlande), FRF (France), GRD (Grèce), IEP (Irlande), ITL (Italie), LUF (Luxembourg), NLG (Pays-Bas) ou PTE (Portugal).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

EUROCONVERT(nombre ; devise source ; devise cible)

##### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Monnaie source, *Type* : Texte

*Commentaire* : Monnaie cible, *Type* : Texte

##### Exemples

EUROCONVERT(1 ; "EUR" ; "DEM") est égal à 1.95583

##### Fonctions en relation

[EURO](#)

#### 8.1.6.21 FV

La fonction FV() renvoie la valeur future d'un investissement, étant donné le rendement et le temps écoulé. Si vous avez 1 000 € sur un compte bancaire gagnant 8 % d'intérêt, après deux années, vous aurez FV(1000;0,08;2), soit 1 166,40 €.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

FV(valeur\_actuelle ; rendement ; périodes)

##### Paramètres

*Commentaire* : Valeur actuelle, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Vitesse, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Périodes, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

FV(1000;0,08;2) vaut 1 166,40

##### Fonctions en relation

[PV](#)

[NPER](#)

[PMT](#)

[RATE](#)

#### 8.1.6.22 FV\_ANNUIITY

La fonction FV\_ANNUIITY() renvoie la valeur future d'un flux de paiements étant donné le montant du paiement, le taux d'intérêt et le nombre de périodes. Par exemple : si vous recevez 500 € par an pendant 20 ans, et l'investissez à 8 %, le total après 20 ans sera FV\_ANNUIITY(500 ; 0,08 ; 20), soit 22 880,98 €. Cette fonction suppose que les paiements sont effectués à la fin de chaque période.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

FV\_ANNUIITY(somme ; intérêt ; période)

### Paramètres

*Commentaire* : Paiement par période, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Taux d'intérêt, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Périodes, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

FV\_ANNUITY(1000; 0,05; 5) vaut 5 525.63

#### 8.1.6.23 INTRATE

La fonction INTRATE renvoie le taux d'intérêt pour un titre complètement investi. "Base" est le type de comptage de jours que vous voulez utiliser : 0 : 30 / 360 US (par défaut), 1 : vrais jours, 2 : vrais jours / 360, 3 : vrais jours / 365 ou 4 : 30 / 365 européen.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

INTRATE(règlement; échéance; investissement; remise, base)

### Paramètres

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Échéance, *Type* : Date

*Commentaire* : Investissement, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Remise, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base de comptage des jours, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

INTRATE("28/2/2001" ; "31/8/2001" ; 1000000 ; 2000000 ; 1) renvoie 1,98

#### 8.1.6.24 IPMT

IPMT calcule le montant de l'annuité allant au remboursement des intérêts.

Taux est le taux d'intérêt périodique.

Période est la période d'amortissement. 1 pour la première période et NPER pour la dernière.

NPER est le nombre total de périodes pendant lequel chaque annuité est payée.

PV est la valeur présente dans la séquence des paiements.

FV (optionnel) est la valeur désirée (future). Par défaut : 0.

Type (optionnel) définit la date de versement. 1 pour le paiement au début de la période et 0 (par défaut) pour le paiement à la fin de la période.

L'exemple montre l'intérêt à payer dans la dernière année d'un prêt sur trois années. Le taux d'intérêt est 10 pour cent.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

IPMT(Taux; Période; NPer; PV; FV; Type)

### Paramètres

*Commentaire* : Vitesse, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Période, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Nombre de périodes, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur actuelle, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur future (optionnel), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Type (optionnel), *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

IPMT(0,1; 3; 3; 8000) vaut -292,45

### Fonctions en relation

PPMT  
PV  
PMT

#### 8.1.6.25 IRR

La fonction IRR calcule le taux interne de renvoi pour une série de cash flows.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

IRR( Valeurs[; Invité = 0.1 ] )

### Paramètres

*Commentaire* : Valeurs, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Invité, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Fonctions en relation

XIRR

#### 8.1.6.26 ISPMT

Calcule l'intérêt payé sur une période donnée d'un investissement.

Taux est le taux d'intérêt périodique.

Période est la période d'amortissement. 1 pour le premier et NPer pour la dernière période.

NPer est le nombre total de périodes durant lesquelles chaque annuité est payée.

PV est la valeur présente dans la séquence des paiements.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

ISPMT(Taux; Période; NPer; PV)

### Paramètres

*Commentaire* : Vitesse, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Période, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombre de périodes, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Valeur actuelle (PV), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

ISPMT(0,1; 1; 3; 8000000) vaut -533 333

### Fonctions en relation

PV  
FV  
NPER  
PMT  
RATE

### 8.1.6.27 LEVEL\_COUPON

La fonction LEVEL\_COUPON() calcule la valeur d'un bon level-coupon. Par exemple : si le taux d'intérêt est 10 %, un bon de 1 000 € avec des coupons semi-annuels à un taux de 13 % qui mature 4 ans vaut LEVEL\_COUPON(1000;0,13;2;4;0,1), soit 1 096.95 €.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

LEVEL\_COUPON(valeur faciale ;taux annueldu coupon;années;taux du marché)

#### Paramètres

*Commentaire* : Valeur faciale, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Taux du coupon, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Coupons par an, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Années, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Taux d'intérêt du marché, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

LEVEL\_COUPON(1000;0,13 ; 2 ; 4;0,1) vaut 1 096.95

### 8.1.6.28 MDURATION

en années

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

MDURATION( règlement ; échéance ; coupon ; rendement ; fréquence ; [ Base=0 ] )

#### Paramètres

*Commentaire* : Règlement, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Échéance, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Coupon, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Rendement, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : La fréquence, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

MDURATION("2004-02-01" ; "2004-05-31" ; 0.08 ; 0.09 ; 2 ; 0) renvoie 0.316321106

#### Fonctions en relation

[DURATION](#)

### 8.1.6.29 MIRR

La fonction MIRR() calculera le pourcentage interne modifié de retour (IRR) d'une série d'investissements périodiques.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

MIRR(valeurs ; investissement ; réinvestissement)

### Paramètres

*Commentaire* : Valeurs, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Investissement, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Réinvestissement, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

MIRR({100;200;-50;300;-200}, 5%, 6%) est égal à 34.2823387842%

### Fonctions en relation

[IRR](#)

#### 8.1.6.30 NOMINAL

La fonction NOMINAL() calcule le taux d'intérêt (fixe) pour un taux d'intérêt effectif (annualisé) composé à des intervalles donnés. Pas exemple, pour gagner 8 % sur un acompte composé mensuellement, vous avez besoin d'un retour de NOMINAL(0,08), soit 7,72 %.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

NOMINAL(effectif;périodes)

### Paramètres

*Commentaire* : Taux effectif d'intérêt, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Périodes, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

NOMINAL(0,08;12) vaut 0,0772

### Fonctions en relation

[EFFECT](#)

#### 8.1.6.31 NPER

Renvoie le nombre de périodes d'un investissement.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

NPER(taux ; paiement ; pv ; fv ; type)

### Paramètres

*Commentaire* : Vitesse, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Paiement, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur actuelle (PV), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur future (FV - optionnel), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Type (optionnel), *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

NPER(0,1 ; -100 ; 1000) vaut 11

### Exemples

NPER(0,06 ; 0 ; -10000 ; 20000 ; 0) renvoie 11 906

### Fonctions en relation

[FV](#)

[RATE](#)

[PMT](#)

[PV](#)



### 8.1.6.32 NPV

La valeur actuelle nette (NPV) pour une série de cash flows périodiques.

Affiche la valeur actuelle nette pour une série de cash flows avec le pourcentage de réduction. Les valeurs peuvent être positives si elles sont reçues comme des recettes et négatives si les sommes sont payées à l'extérieur.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

NPV(taux;valeurs)

#### Paramètres

*Commentaire* : Vitesse, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs (tableau), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

NPV(100%;4;5;7) = 4.125

#### Fonctions en relation

FV

IRR

NPER

PMT

PV

### 8.1.6.33 ODDLPRICE

La fonction ODDLPRICE calcule la valeur de la sécurité par 100 unités courantes de valeur faciale. La sécurité a une date irrégulière de dernier intérêt.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

ODDLPRICE( règlement; échéance; dernier; taux; rendementannuel; remise; fréquence [; Base = 0 ] )

#### Paramètres

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Échéance, *Type* : Date

*Commentaire* : Dernier, *Type* : Date

*Commentaire* : Vitesse, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : AnnualYield, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Remise, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : La fréquence, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

ODDLPRICE( DATE(1990;6;1); DATE(1995;12;31); DATE(1990;1;1); 3%; 5%; 100; 2) renvoie 90.991042345

### 8.1.6.34 ODDLYIELD

La fonction ODDLYIELD calcule le rendement de la sécurité qui a une date de dernier intérêt irrégulière.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

ODDLYIELD( règlement ; échéance ; dernier ; taux ; prix ; remise ; fréquence [ ; Base = 0 ] )

### Paramètres

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Échéance, *Type* : Date

*Commentaire* : Dernier, *Type* : Date

*Commentaire* : Vitesse, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Prix, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Remise, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : La fréquence, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

ODDLYIELD( DATE(1990;6;1); DATE(1995;12;31); DATE(1990;1;1); 3%; 91; 100; 2) renvoie 4.997775351

### Fonctions en relation

[ODDLPRICE](#)

### 8.1.6.35 PMT

PMT renvoie le montant du paiement d'un prêt utilisant un taux d'intérêt fixe et des versements constants (chaque versement est d'un montant égal).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

PMT( taux ; nper ; pv [ ; fv = 0 [ ; type = 0 ] ] )

### Paramètres

*Commentaire* : Vitesse, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Nombre de périodes (NPer), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur actuelle (PV), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur future (FV - optionnel), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Type (optionnel), *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

PMT(0,1 ; 4 ; 10000) vaut -3 154,71

### Fonctions en relation

[NPER](#)

[IPMT](#)

[PPMT](#)

[PV](#)

### 8.1.6.36 PPMT

PPMT calcule le montant de l'annuité allant au remboursement du principal.

Taux est le taux d'intérêt périodique.

Période est la période d'amortissement. 1 pour la première période et NPER pour la dernière.

NPER est le nombre total de périodes pendant lequel chaque annuité est payée.

PV est la valeur présente dans la séquence des paiements.

FV (optionnel) est la valeur désirée (future). Par défaut : 0.

Type (optionnel) définit la date de versement. 1 pour le paiement au début de la période et 0 (par défaut) pour le paiement à la fin de la période.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

PPMT(Taux; Période; NPer; PV [; FV = 9 [; Type = 0 ]])

### Paramètres

*Commentaire* : Vitesse, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Période, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Nombre de périodes, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur actuelle, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur future (optionnel), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Type (optionnel), *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

PPMT(0,0875; 1; 36; 5000; 8000; 1) vaut -18,48

### Fonctions en relation

[IPMT](#)

[PMT](#)

[PV](#)

### 8.1.6.37 PRICEMAT

PRICEMAT calcule le prix par 100 unités courantes de valeur faciale de la sécurité qui produit intérêt pour la date d'échéance.

Méthode de calcul de base

0 méthode US, 12 mois, chaque mois avec 30 jours

1 Nombre actuel de jours dans l'année, nombre actuel de jours dans les mois

2 360 jours dans une année, nombre actuel de jours dans les mois

4 365 jours dans une année, nombre actuel de jours dans les mois

5 Méthode européenne, 12 mois, chaque mois a 30 jours

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

PRICEMAT(règlement; échéance; sortie; taux; rendement; [; base = 0 ])

### Paramètres

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Échéance, *Type* : Date

*Commentaire* : Sortie, *Type* : Date

*Commentaire* : Taux de rabais, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Rendement, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

PRICEMAT(DATE(1990;6;1);DATE(1995;12;31);DATE(1990;1;1);6%;5%)      renvoie  
103.819218241

### 8.1.6.38 PV

La fonction PV() renvoie la valeur présente d'un investissement -- la valeur actuelle d'une somme d'argent dans l'avenir, étant donné le taux d'intérêt ou l'inflation. Par exemple, si vous avez besoin de 1 166,40 € pour votre nouvel ordinateur et que vous voulez l'acheter dans deux ans avec un taux d'intérêt de 8 %, vous avez besoin de commencer avec PV(1166,4;0,08;2), soit 1 000 €.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

PV(valeur\_future; intérêt; périodes)

#### Paramètres

*Commentaire* : Valeur future, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Taux d'intérêt, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Périodes, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

PV(1166,4; 0,08; 2) vaut 1 000

### 8.1.6.39 PV\_ANNUIITY

La fonction PV\_ANNUIITY() renvoie la valeur présente d'une annuité ou flux de paiements. Par exemple : un ticket de loterie d'un million d'euros qui paie 50 000 € par an pendant 20 ans, avec un intérêt de 5 %, vaut actuellement PV\_annuity(50000; 0,05; 20), soit 623 111 €. Cette fonction suppose que les paiements sont effectués à la fin de chaque période.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

PV\_ANNUIITY(montant; intérêt; périodes)

#### Paramètres

*Commentaire* : Paiement par période, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Taux d'intérêt, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Périodes, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

PV\_ANNUIITY(1000; 0,05; 5) vaut 4 329.48

### 8.1.6.40 RATE

La fonction RATE() calcule le taux d'intérêt constant par période d'un investissement.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

RATE(nper;pmt;pv;fv;type;guess)

#### Paramètres

*Commentaire* : Période de remboursement, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Versements périodiques, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur actuelle, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur future, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Type, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Invité, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

RATE(4\*12;-200;8000) vaut 0.007701472

#### 8.1.6.41 RECEIVED

La fonction RECEIVED renvoie la somme reçue à la date d'échéance pour un titre. "Base" est le type de comptage de jours que vous voulez utiliser : 0 : 30 / 360 US (par défaut), 1 : vrais jours, 2 : vrais jours / 360, 3 : vrais jours / 365 ou 4 : 30 / 365 européen. La date de règlement doit être avant la date d'échéance.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

RECEIVED(règlement;échéance;investissement;remise;base)

#### Paramètres

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Échéance, *Type* : Date

*Commentaire* : Investissement, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Taux de rabais, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

RECEIVED("28/2/2001"; "31/8/2001"; 1000; 0,05; 0) renvoie 1,025.787

#### 8.1.6.42 RRI

La fonction RRI calcule le taux d'intérêt résultant d'un retour (profit) d'investissement.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

RRI( P ; Pv ; Fv)

#### Paramètres

*Commentaire* : P, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Pv, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Fv, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

RRI(1;100;200) renvoie 1

#### Fonctions en relation

FV  
NPER  
PMT  
PV  
RATE

#### 8.1.6.43 SLN

La fonction SLN() détermine la dépréciation linéaire d'un bien pour une période. "Coût" est la somme que vous avez payée pour le bien. "Résidu" est la valeur du bien à la fin de la période. "Vie" est le nombre de périodes pendant lesquelles il perd de la valeur. "SLN" divise le coût régulièrement pendant la vie du bien.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

SLN(coût;valeur résiduelle; vie)

### Paramètres

*Commentaire* : Coût, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Résidu, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Vie, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

SLN(10000;700;10) vaut 930

### Fonctions en relation

SYD

DDB

#### 8.1.6.44 SYD

La fonction SYD() calcule la dépréciation annuelle chiffrée pour un bien en fonction de son coût, de la valeur résiduelle, de la vie prévue et d'une période particulière. Cette méthode accélère le taux de dépréciation, ainsi une plus grande dépréciation survient au début plutôt qu'à la fin. Le coût de dépréciation est la valeur actuelle moins la valeur résiduelle. La vie utile est le nombre de périodes (typiquement des années) sur lesquelles le bien est déprécié.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

SYD(coût; valeur résiduelle; vie; période)

### Paramètres

*Commentaire* : Coût, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Résidu, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Vie, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Période, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

SYD(5000; 200; 5; 2) vaut 1280

### Fonctions en relation

SLN

DDB

#### 8.1.6.45 TBILLEQ

La fonction TBILLEQ renvoie l'équivalent de bon pour une note de trésorerie. La date d'échéance doit être après la date de règlement, mais dans les 365 jours.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

TBILLEQ(règlement; échéance; remise)

### Paramètres

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Échéance, *Type* : Date

*Commentaire* : Taux de rabais, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

TBILLEQ("28/2/2001"; "31/8/2001"; 0,1) renvoie 0,1068

### Fonctions en relation

TBILLPRICE

TBILLYIELD

#### 8.1.6.46 TBILLPRICE

La fonction TBILLPRICE renvoie le prix par valeur de 100 € pour une note de trésorerie. La date d'échéance doit être après la date de déclaration, mais dans les 365 jours. Le taux d'escompte doit être positif.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

TBILLPRICE(règlement;échéance;remise)

##### Paramètres

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Échéance, *Type* : Date

*Commentaire* : Taux de rabais, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

TBILLPRICE("28/2/2001";"31/8/2001";0,05) renvoie 97,444

##### Fonctions en relation

[TBILLEQ](#)

[TBILLYIELD](#)

#### 8.1.6.47 TBILLYIELD

La fonction TBILLYIELD renvoie le rendement pour une note de trésorerie. La date d'échéance doit être après la date de règlement mais dans les 365 jours. Le prix doit être positif.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

TBILLYIELD(règlement;échéance;prix)

##### Paramètres

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Échéance, *Type* : Date

*Commentaire* : Prix par valeur faciale de 100 €, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

TBILLYIELD("28/2/2001";"31/8/2001";600) renvoie -1.63

##### Fonctions en relation

[TBILLEQ](#)

[TBILLPRICE](#)

#### 8.1.6.48 VDB

VDB calcule la dépréciation d'un bien avec une valeur initiale, une vie utile espérée, et une valeur résiduelle finale, pour une période donnée, en utilisant la méthode de balance à déclin fixé.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

VDB(coût; résidu; vie; début-période; fin-période; [, facteur-dépréciation = 2 [, changer = faux ]])

##### Paramètres

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Échéance, *Type* : Date

*Commentaire* : Prix, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Remise, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

VDB(10000;600;10;0;0.875;1.5) renvoie 1312.5

#### 8.1.6.49 XIRR

La fonction XIRR calcule le taux interne de renvoi pour une série non-périodique de cash flows.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

XIRR( Valeurs; Dates[; Invité = 0.1 ] )

##### Paramètres

*Commentaire* : Valeurs, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Dates, *Type* : Date

*Commentaire* : Invité, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

XIRR(B1 :B4;C1 :C4) suppose que B1 :B4 contienne -20000, 4000, 12000, 8000 pendant que C1 :C4 contient "=DATE(2000;1;1)", "=DATE(2000;6;1)", "=DATE(2000;12;30)", "=DATE(2001;3;1)" renvoie 0.2115964

##### Fonctions en relation

[IRR](#)

#### 8.1.6.50 XNPV

La fonction XNPV calcule la valeur présente actuelle d'une série de cash flows.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

XNPV( Taux; Valeurs; Dates )

##### Paramètres

*Commentaire* : Vitesse, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Dates, *Type* : Date

##### Exemples

XNPV(5%;B1 :B4;C1 :C4) suppose que B1 :B4 contienne -20000, 4000, 12000, 8000 pendant que C1 :C4 contient "=DATE(2000;1;1)", "=DATE(2000;6;1)", "=DATE(2000;12;30)", "=DATE(2001;3;1)" renvoie 2907.83187

##### Fonctions en relation

[NPV](#)

#### 8.1.6.51 YIELDDISC

YIELDDISC calcule le rendement d'une sécurité décomptée par 100 unités courantes de valeur faciale.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

YIELDDISC(règlement; échéance; prix, remise, base)

##### Paramètres

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Échéance, *Type* : Date

*Commentaire* : Prix, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Remise, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

YIELDDISC(DATE(1990;6;1);DATE(1990;12;31);941.66667;1000) renvoie 0.106194684



### 8.1.6.52 YIELDMAT

La fonction YIELDMAT calcule le seuil de sécurité significatif pour la date d'échéance.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

YIELDMAT(règlement;échéance;terme;taux;prix;base)

#### Paramètres

*Commentaire* : Règlement, *Type* : Date

*Commentaire* : Échéance, *Type* : Date

*Commentaire* : Sortie, *Type* : Date

*Commentaire* : Taux de rabais, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Prix, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

YIELDMAT(DATE(1990;6;1);DATE(1995;12;31);DATE(1990;1;1);6%;103.819218241) renvoie 0.050000000

#### Fonctions en relation

[YIELDDISC](#)

### 8.1.6.53 ZERO\_COUPON

La fonction ZERO\_COUPON() calcule la valeur d'un bon zéro-coupon (pure réduction). Par exemple : si le taux d'intérêt est 10 %, un bon de 1 000 € qui mature 20 ans vaut ZERO\_COUPON(1000;0,1;20), soit 148,64 €.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

ZERO\_COUPON(valeur faciale;taux;années)

#### Paramètres

*Commentaire* : Valeur faciale, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Taux d'intérêt, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Années, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

ZERO\_COUPON(1000;0,1;20) vaut 148,64

## 8.1.7 Information

### 8.1.7.1 ERRORTYPE

La fonction ERRORTYPE() convertit une erreur en nombre. Si la valeur n'est pas une erreur, une erreur est retournée. sinon, un code numérique est retourné. les codes d'erreur sont calqués sur Excel.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

#### Syntaxe

ERRORTYPE(valeur)

#### Paramètres

*Commentaire* : Erreur, *Type* : N'importe quel type de valeur

#### Exemples

ERRORTYPE(NA()) renvoie 7

#### Exemples

ERRORTYPE(0) renvoie une erreur

#### 8.1.7.2 FILENAME

Renvoie le nom du fichier courant. Si le document courant n'est pas enregistré, une chaîne vide est renvoyée.

*Type de retour* : Texte

##### Syntaxe

FILENAME()

##### Paramètres

#### 8.1.7.3 FORMULA

La fonction FORMULA() renvoie la d'une cellule comme une chaîne.

*Type de retour* : Texte

##### Syntaxe

FORMULA(x)

##### Paramètres

*Commentaire* : Référence, *Type* : Référence

##### Exemples

FORMULA(A1) retourne "=SUM(1+2)" si la cellule A1 contient une telle formule

#### 8.1.7.4 INFO

La fonction INFO() renvoie des informations sur l'environnement courant. Le type de paramètre spécifie quel type d'information vous voulez recevoir. C'est l'un parmi les suivants : "directory" renvoie l'emplacement du dossier courant, "numfile" renvoie le nombre de documents actifs, "release" renvoie la version de Calligra Sheets sous forme de texte, "recalc" renvoie le mode courant de recalcul : "Automatic" ou "Manual", "system » renvoie le nom de l'environnement d'exploitation, "osversion" renvoie le système d'exploitation courant.

*Type de retour* : Texte

##### Syntaxe

INFO(type)

##### Paramètres

*Commentaire* : Type d'information, *Type* : Texte

#### 8.1.7.5 ISBLANK

La fonction ISBLANK() renvoie "Vrai" si le paramètre est vide. Sinon elle renvoie "Faux".

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISBLANK(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Exemples

ISBLANK(A1) renvoie "Vrai" si A1 est vide

##### Exemples

ISBLANK(A1) renvoie "Faux" si A1 contient une valeur

#### 8.1.7.6 ISDATE

La fonction ISDATE() renvoie "Vrai" si le paramètre est une date, sinon elle renvoie "Faux".

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISDATE(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Exemples

ISDATE("2/2/2000") renvoie Vrai

##### Exemples

ISDATE("salut") renvoie Faux

#### 8.1.7.7 ISERR

La fonction ISERR() renvoie "Vrai" si le paramètre est une erreur autre que "N/A", sinon elle renvoie "Faux". Utilisez la fonction ISERROR() si vous voulez inclure "N/A".

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISERR(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Fonctions en relation

[ISERROR](#)

[ISNA](#)

#### 8.1.7.8 ISERROR

La fonction ISERROR() renvoie "Vrai" si le paramètre est une erreur de n'importe quel type, sinon elle renvoie "Faux".

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISERROR(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Fonctions en relation

[ISERR](#)

[ISNA](#)

#### 8.1.7.9 ISEVEN

La fonction ISEVEN() renvoie "Vrai" si le nombre est pair, sinon elle renvoie "Faux".

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISEVEN(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Exemples

ISEVEN(12) renvoie Vrai

##### Exemples

ISEVEN(-7) renvoie Faux

#### 8.1.7.10 ISFORMULA

La fonction ISFORMULA() renvoie Vrai si la cellule référencée contient une formule. Sinon, elle renvoie faux.

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISFORMULA(x)

##### Paramètres

*Commentaire* : Référence, *Type* : Référence

#### 8.1.7.11 ISLOGICAL

La fonction ISLOGICAL() renvoie "Vrai" si le paramètre est une valeur booléenne. Sinon elle renvoie "Faux".

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISLOGICAL(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Exemples

ISLOGICAL(A1 >A2) renvoie "Vrai"

##### Exemples

ISLOGICAL(12) renvoie "Faux"

#### 8.1.7.12 ISNA

La fonction ISNA() renvoie "Vrai" si le paramètre est une erreur "N/A", sinon elle renvoie "Faux".

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISNA(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Fonctions en relation

[ISERR](#)

[ISERROR](#)

#### 8.1.7.13 ISNONTEXT

La fonction ISNONTEXT() renvoie "Vrai" si le paramètre n'est pas une chaîne, sinon elle renvoie "Faux". Elle agit comme ISNOTTEXT.

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISNONTEXT(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Exemples

ISNONTEXT(12) renvoie Vrai

##### Exemples

ISNONTEXT("HELLO") renvoie Faux

##### Fonctions en relation

[ISNOTTEXT](#)

#### 8.1.7.14 ISNOTTEXT

La fonction ISNOTTEXT() renvoie "Vrai" si le paramètre n'est pas une chaîne, sinon elle renvoie "Faux". Elle agit comme ISNONTEXT.

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISNOTTEXT(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Exemples

ISNOTTEXT(12) renvoie Vrai

##### Exemples

ISNOTTEXT("salut") renvoie Faux

##### Fonctions en relation

[ISNONTEXT](#)

#### 8.1.7.15 ISNUM

La fonction ISNUM() renvoie "Vrai" si le paramètre est une valeur numérique, sinon elle renvoie "Faux". Elle agit comme ISNUMBER.

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISNUM(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Exemples

ISNUM(12) renvoie Vrai

##### Exemples

ISNUM(salut) renvoie Faux

##### Fonctions en relation

[ISNUMBER](#)

#### 8.1.7.16 ISNUMBER

La fonction ISNUMBER() renvoie "Vrai" si le paramètre est une valeur numérique, sinon elle renvoie "Faux". Elle agit comme ISNUM.

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISNUMBER(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Exemples

ISNUMBER(12) renvoie Vrai

##### Exemples

ISNUMBER("salut") renvoie Faux

##### Fonctions en relation

[ISNUM](#)

#### 8.1.7.17 ISODD

La fonction ISODD() renvoie "Vrai" si le nombre est impair, sinon elle renvoie "Faux".

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISODD(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Exemples

ISODD(12) renvoie Faux

##### Exemples

ISODD(-7) renvoie Vrai

#### 8.1.7.18 ISREF

La fonction ISREF() renvoie "Vrai" si le paramètre se rapporte à une référence, sinon elle renvoie "Faux"

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISREF(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Exemples

ISREF(A12) renvoie Vrai

##### Exemples

ISREF("salut") renvoie Faux

#### 8.1.7.19 ISTEEXT

La fonction ISTEEXT() renvoie "Vrai" si le paramètre est une chaîne, sinon elle renvoie "Faux"

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISTEEXT(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Exemples

ISTEEXT(12) renvoie Faux

##### Exemples

ISTEEXT("HELLO") renvoie Vrai

#### 8.1.7.20 ISTEME

La fonction ISTEME() renvoie "Vrai" si le paramètre est une valeur de temps, sinon elle renvoie "Faux".

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

ISTEME(x)

##### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

##### Exemples

ISTEME("12 :05") renvoie Vrai

##### Exemples

ISTEME("salut") renvoie Faux

#### 8.1.7.21 N

La fonction N() convertit une valeur en un nombre. Si une valeur est ou se rapporte à un nombre, cette fonction renvoie le nombre. Si la valeur est "Vrai", cette fonction renvoie 1. Si une valeur est une date, cette fonction renvoie le nombre de série de la date. Toute autre chose renverra 0.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

N(Valeur)

##### Paramètres

*Commentaire* : Valeur, *Type* : N'importe quel type de valeur

##### Exemples

N(3,14) renvoie 3,14

##### Exemples

N("7") renvoie 0 (car "7" est du texte)

#### 8.1.7.22 NA

La fonction NA() renvoie la valeur d'erreur permanente, N/A.

*Type de retour* : Erreur

##### Syntaxe

NA()

##### Paramètres

##### Fonctions en relation

[ISNA](#)

[ISERR](#)

[ISERROR](#)

#### 8.1.7.23 TYPE

La fonction TYPE() renvoie 1 si la valeur est un nombre, 2 si c'est du texte, 4 si la valeur est une valeur logique, 16 si c'est une valeur d'erreur ou 64 si la valeur est un tableau. Si la cellule que la valeur représente contient une formule, vous obtenez son type de retour.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

### Syntaxe

TYPE(x)

### Paramètres

*Commentaire* : Toute valeur, *Type* : N'importe quel type de valeur

### Exemples

TYPE(A1) renvoie 2 si A1 contient du texte

### Exemples

TYPE(-7) renvoie 1

### Exemples

TYPE(A2) renvoie 1, si A2 contient "=CURRENTDATE()"

## 8.1.8 Logiques

### 8.1.8.1 AND

La fonction AND() renvoie "Vrai" si toutes les valeurs sont vraies, sinon elle renvoie "Faux" (au moins une des valeurs dans une erreur renvoyées).

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

### Syntaxe

AND(valeur;valeur;...)

### Paramètres

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

### Exemples

AND(vrai;vrai;vrai) renvoie Vrai

### Exemples

AND(Vrai;Faux) renvoie Faux

### 8.1.8.2 FALSE

La fonction FALSE() renvoie la valeur Booléenne Faux.

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

### Syntaxe

FALSE()

### Paramètres

### Exemples

FALSE() renvoie Faux

### 8.1.8.3 IF

La fonction IF() est une fonction de condition. Cette fonction renvoie le deuxième paramètre si la condition est Vrai, sinon elle renvoie le troisième paramètre (qui par défaut est faux).

*Type de retour* : N'importe quel type de valeur



### Syntaxe

IF(condition;si\_vrai;si\_faux)

### Paramètres

*Commentaire* : Condition, *Type* : Une valeur de vérité (VRAI ou FAUX)

*Commentaire* : Si\_vrai, *Type* : N'importe quel type de valeur

*Commentaire* : Si\_faux, *Type* : N'importe quel type de valeur

### Exemples

A1=4, A2=6, IF(A1 >A2;5;3) renvoie 3

#### 8.1.8.4 IFERROR

Renvoie X, à moins qu'il y ait une erreur. Dans ce cas, renvoie une valeur alternative.

*Type de retour* : N'importe quel type de valeur

### Syntaxe

IFERROR(AnyX;AnyAlternative)

### Paramètres

*Commentaire* : Any X, *Type* : N'importe quel type de valeur

*Commentaire* : Toute alternative, *Type* : N'importe quel type de valeur

### Exemples

IFERROR(A1;A2) renvoie le contenu de A1 s'il ne s'agit pas d'une valeur-erreur, sinon, renvoie de contenu de A2.

#### 8.1.8.5 IFNA

Renvoie X, à moins qu'il y ait une erreur. Dans ce cas, renvoie une valeur alternative.

*Type de retour* : N'importe quel type de valeur

### Syntaxe

IFNA(AnyX;AnyAlternative)

### Paramètres

*Commentaire* : Any X, *Type* : N'importe quel type de valeur

*Commentaire* : Toute alternative, *Type* : N'importe quel type de valeur

### Exemples

IFNA(A1;A2) renvoie le contenu de A1 s'il ne s'agit pas d'une valeur-erreur #NA, sinon, renvoie de contenu de A2.

#### 8.1.8.6 NAND

La fonction NAND() renvoie "Vrai" si au moins une valeur n'est pas vraie, sinon elle renvoie "Faux".

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

### Syntaxe

NAND(valeur;valeur;...)

### Paramètres

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

### Exemples

NAND(Vrai;Faux;Faux) renvoie Vrai

### Exemples

NAND(Vrai;Vrai) renvoie Faux

### 8.1.8.7 NOR

La fonction NOR() renvoie "Vrai" si toutes les valeurs données en paramètres sont de type booléen et ont la valeur "Faux". Sinon, elle renvoie "Faux".

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

#### Syntaxe

NOR(valeurs;valeurs;...)

#### Paramètres

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

#### Exemples

NOR(Vrai;Faux;Faux) renvoie Faux

#### Exemples

NOR(Faux;Faux) renvoie Vrai

### 8.1.8.8 NOT

La fonction NOT() renvoie "Vrai" si la valeur est fausse et "Faux" si la valeur est "Vrai". Elle renvoie une erreur si au moins une entrée est une erreur.

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

#### Syntaxe

NOT(bool)

#### Paramètres

*Commentaire* : Valeur booléenne, *Type* : Une valeur de vérité (VRAI ou FAUX)

#### Exemples

NOT(Faux) renvoie Vrai

#### Exemples

NOT(Vrai) renvoie Faux

### 8.1.8.9 OR

La fonction OR() renvoie "Vrai" si au moins une valeur est vraie, sinon elle renvoie "Faux" (au moins une des valeurs dans une erreur renvoyées).

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

#### Syntaxe

OR(valeur;valeur;...)

#### Paramètres

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

#### Exemples

OR(Faux;Faux;Faux) renvoie Faux

#### Exemples

OR(Vrai;Faux) renvoie Vrai

#### 8.1.8.10 TRUE

La fonction TRUE() renvoie la valeur booléenne Vrai.

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

TRUE()

##### Paramètres

##### Exemples

TRUE() renvoie Vrai

#### 8.1.8.11 XOR

La fonction XOR() renvoie "Vrai" si le nombre de valeurs vraies est pair, sinon elle renvoie "Faux". Elle renvoie une erreur si au moins un argument est une erreur.

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

XOR(valeur;valeur;...)

##### Paramètres

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

*Commentaire* : Valeurs booléennes, *Type* : Une plage de valeurs de vérité (VRAI ou FAUX)

##### Exemples

XOR(Faux;Faux;Faux) renvoie Vrai

##### Exemples

OR(Vrai;Faux) renvoie Vrai

### 8.1.9 Recherche et références

#### 8.1.9.1 ADDRESS

ADDRESS crée une adresse de cellule. Le paramètre Ligne est le numéro de ligne et Colonne est le numéro de colonne.

Le nombre absolu spécifie le type de référence : 1 ou omis = Absolu, 2 = ligne absolue, colonne relative, 3 = ligne relative, colonne absolue et 4 = Relatif.

Style A1 spécifie le style de l'adresse à retourner. Si A1 est positionné sur Vrai (par défaut), l'adresse est retournée en style A1, s'il est positionné sur Faux, en style R1C1.

Le nom de la feuille est le texte qui spécifie le nom de la feuille.

*Type de retour* : Texte

##### Syntaxe

ADDRESS(ligne ; col ; absolue ; style ; nom\_de\_feuille)

##### Paramètres

*Commentaire* : Numéro de ligne, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Numéro de colonne, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombre absolu (optionnel), *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Style A1 (optionnel), *Type* : Une valeur de vérité (VRAI ou FAUX)

*Commentaire* : Nom de la feuille, *Type* : Texte

**Exemples**

ADDRESS(6; 4) renvoie  $\$D\$6$

**Exemples**

ADDRESS(6; 4; 2) renvoie  $D\$6$

**Exemples**

ADDRESS(6; 4; 2; Faux; "Feuille1") renvoie Feuille1!R6C[4]

**Exemples**

ADDRESS(6; 4; 1; Faux; "Feuille1") renvoie Feuille1!R6C4

**Exemples**

ADDRESS(6; 4; 4; Vrai; "Feuille1") renvoie Feuille1!D6

### 8.1.9.2 AREAS

Renvoie le nombre de zones dans la chaîne de référence. Une zone peut être une seule cellule ou un lot de cellules.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

**Syntaxe**

AREAS(référence)

**Paramètres**

*Commentaire* : Référence, *Type* : Une plage de chaînes

**Exemples**

AREAS(A1) renvoie 1

**Exemples**

AREAS((A1; A2 :A4)) renvoie 2

### 8.1.9.3 CELL

Renvoie des informations sur la position, le formatage et le contenu dans une référence.

*Type de retour* : N'importe quel type de valeur

**Syntaxe**

CELL(type; référence)

**Paramètres**

*Commentaire* : Type, *Type* : Texte

*Commentaire* : Référence, *Type* : Référence

**Exemples**

CELL("COL", C7) renvoie 3

**Exemples**

CELL("ROW", C7) renvoie 7

**Exemples**

CELL("ADDRESS", C7) renvoie  $\$C\$7$

#### 8.1.9.4 CHOOSE

Renvoie le paramètre spécifié par l'index.

*Type de retour* : N'importe quel type de valeur

##### Syntaxe

CHOOSE(index; paramètre1; paramètre2,...)

##### Paramètres

*Commentaire* : Indice, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Arguments, *Type* :

##### Exemples

CHOOSE(1; "1er"; "2em") renvoie "1er"

##### Exemples

CHOOSE(2; 3; 2; 4) renvoie 2

#### 8.1.9.5 COLUMN

La fonction COLUMN renvoie la colonne de la cellule donnée en référence. Si aucun paramètre n'est spécifié, la colonne de la cellule courante est renvoyée.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

COLUMN(référence)

##### Paramètres

*Commentaire* : Référence, *Type* : Texte

##### Exemples

COLUMN(A1) renvoie 1

##### Exemples

COLUMN(D2) renvoie 4

##### Fonctions en relation

[COLUMNS](#)

[ROW](#)

#### 8.1.9.6 COLUMNS

La fonction COLUMNS renvoie le nombre de colonnes dans une référence.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

COLUMNS(référence)

##### Paramètres

*Commentaire* : Référence, *Type* : Texte

##### Exemples

COLUMNS(A1 :C3) renvoie 3

##### Exemples

COLUMNS(D2) renvoie 1

##### Fonctions en relation

[COLUMN](#)

[ROWS](#)

### 8.1.9.7 HLOOKUP

Cherche une valeur correspondante dans la table donnée, et renvoie la valeur de la ligne indiquée.

Regarder la valeur de recherche dans la première ligne de la base de données. Si une valeur correspond, la valeur dans la ligne et la colonne, la valeur trouvée, est renvoyée. Si le tri est vrai (par défaut), la première ligne est supposée être triée. La recherche finira, si la valeur de recherche est inférieure à la valeur actuellement comparée.

*Type de retour* : Chaîne / numérique

#### Syntaxe

HLOOKUP(valeur de recherche ; base de données ; ligne ; méthode de tri)

#### Paramètres

*Commentaire* : Valeur de recherche, *Type* : Chaîne / numérique

*Commentaire* : Base de données, *Type* : Matrice

*Commentaire* : Row, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Méthode de tri(optionnel), *Type* : Une valeur de vérité (VRAI ou FAUX)

### 8.1.9.8 INDEX

Si une plage est indiquée, la valeur retournée est conservée dans la ligne ou colonne indiquée. Si une cellule contenant un tableau est indiquée, seul le premier élément du tableau est renvoyé.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

#### Syntaxe

INDEX(cellule, ligne, colonne)

#### Paramètres

*Commentaire* : Référence, *Type* : Texte

*Commentaire* : Row, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Colonne, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

INDEX(A1 :C3;2;2) renvoie le contenu de B2

#### Exemples

INDEX(A1;2;2), si A1 est le résultat d'un calcul de tableau, renvoie l'élément (2,2) de ce tableau.

### 8.1.9.9 INDIRECT

Renvoie le contenu d'une cellule spécifiée par le texte de référence. Le deuxième paramètre est optionnel.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

#### Syntaxe

INDIRECT(Texte\_de\_référence,style a1)

#### Paramètres

*Commentaire* : Référence, *Type* : Texte

*Commentaire* : Style A1 (optionnel), *Type* : Une valeur de vérité (VRAI ou FAUX)

#### Exemples

INDIRECT(A1), A1 contient "B1", et B1 1 = > renvoie 1

#### Exemples

INDIRECT("A1"), renvoie le contenu de A1

#### 8.1.9.10 LOOKUP

La fonction LOOKUP cherche le premier paramètre dans le vecteur de recherche. Elle renvoie une valeur dans le vecteur résultat, en conservant les index du vecteur initial. Si la valeur est introuvable, une erreur est renvoyée. Le vecteur de recherche doit être dans l'ordre croissant et les deux doivent avoir la même taille. Les valeurs numériques, les chaînes et les booléens sont reconnus. La comparaison de chaînes n'est pas sensible à la casse.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

LOOKUP(valeur; vecteur de recherche; vecteur résultat)

##### Paramètres

*Commentaire* : Valeur de recherche, *Type* : Chaîne / numérique

*Commentaire* : Vecteur de recherche, *Type* : Chaîne / numérique

*Commentaire* : Vecteur résultat, *Type* : Chaîne / numérique

##### Exemples

LOOKUP(1.232; A1 :A6; B1 :B6) avec A1 = 1, A2 = 2 renvoie la valeur de B1.

#### 8.1.9.11 MATCH

Trouve une valeur de recherche dans une zone de recherche et retourne sa position (à partir de 1). Le type de correspondance peut-être soit -1, 0 ou 1 et détermine quelle est la recherche pour la valeur. Si le type de correspondance est 0, l'index de la première valeur qui est égale à la valeur de recherche est retournée, si le type de correspondance est 1 (ou omis), l'index de la première valeur qui est inférieure ou égale à la valeur de recherche est retourné et les valeurs dans la zone de recherche doit être triée en ordre ascendant. Si le type de correspondance est -1, la plus petite valeur qui est supérieure ou égale à la valeur de recherche est trouvée et la zone de recherche doit être triée en ordre décroissant.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

MATCH(valeur de recherche; zone de recherche : type de correspondance)

##### Paramètres

*Commentaire* : Recherche de valeur, *Type* : Chaîne / numérique

*Commentaire* : Zone de recherche ;, *Type* : Référence / tableau

*Commentaire* : Type de correspondance (optionnel), *Type* : Un nombre entier (comme 1, 132, 2344)

#### 8.1.9.12 MULTIPLE.OPERATIONS

MULTIPLE.OPERATIONS exécute la formule définie par FormulaCell et toutes les expressions dont elle dépend en remplaçant toutes les références à RowCell avec celles à RowReplacement; également en remplaçant toutes les références à ColumnCell avec celles à ColumnReplacement. La fonction peut être utilisée pour créer facilement des tables d'expressions qui dépendent de l'entrée de deux paramètres.

*Type de retour* : Chaîne / numérique

##### Syntaxe

MULTIPLE.OPERATIONS(Formula cell; Row cell; Row replacement; Column cell; Column replacement)

##### Paramètres

*Commentaire* : Formules de cellule, *Type* : Référence

*Commentaire* : Cellule de ligne, *Type* : Référence

*Commentaire* : Remplacement de ligne, *Type* : Référence

*Commentaire* : Cellule de colonne optionnel), *Type* : Référence

*Commentaire* : Remplacement de cellule (optionnel), *Type* : Référence

#### 8.1.9.13 OFFSET

Modifie la position et la dimension d'une référence.

*Type de retour* : Référence

##### Syntaxe

OFFSET(Référence reference; entier offsetLigne; entier offsetColonne; entier nouvelleHaut-  
teur; entier nouvelleLargeur)

##### Paramètres

*Commentaire* : Référence ou plage, *Type* : Référence

*Commentaire* : Nombre de colonnes à décaler, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombre de lignes à décaler, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : hauteur de la plage de décalage (optionnel), *Type* : Un nombre entier (comme  
1, 132, 2344)

*Commentaire* : largeur de la plage de décalage (optionnel), *Type* : Un nombre entier (comme  
1, 132, 2344)

#### 8.1.9.14 ROW

La fonction ROW renvoie la ligne d'une cellule donnée en référence. Si aucun paramètre n'est  
spécifié, la ligne de la cellule courante est renvoyée.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

ROW(référence)

##### Paramètres

*Commentaire* : Référence, *Type* : Texte

##### Exemples

ROW(A1) renvoie 1

##### Exemples

ROW(D2) renvoie 2

##### Fonctions en relation

[ROWS](#)

[COLUMN](#)

#### 8.1.9.15 ROWS

La fonction ROWS() renvoie le nombre de lignes dans une référence.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

ROWS(référence)

##### Paramètres

*Commentaire* : Référence, *Type* : Texte

##### Exemples

ROWS(A1 :C3) renvoie 3

##### Exemples

ROWS(D2) renvoie 1

##### Fonctions en relation

[ROW](#)

[COLUMNS](#)



#### 8.1.9.16 SHEET

Renvoie le numéro de la feuille de la référence ou la chaîne représentant un nom de feuille.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

SHEET(référence)

##### Paramètres

*Commentaire* : Référence, *Type* : Référence

##### Exemples

SHEET(Sheet1!C7) renvoie 1

##### Exemples

SHEET(Sheet2!C7) renvoie 2

#### 8.1.9.17 SHEETS

Renvoie le nombre de feuilles dans une référence ou dans le document actuel.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

SHEETS(référence)

##### Paramètres

*Commentaire* : Référence, *Type* : Référence

#### 8.1.9.18 VLOOKUP

Chercher une valeur correspondante dans la première colonne du tableau donné, et renvoyer la valeur de la colonne indiquée.

Regarder la valeur de recherche dans la première colonne de la base de données. Si une valeur correspond, la valeur dans la colonne et la ligne, la valeur trouvée, est renvoyée. Si le tri est vrai (par défaut), la première colonne est supposée être triée. La recherche finira, si la valeur de recherche est inférieure à la valeur actuellement comparée.

*Type de retour* : Chaîne / numérique

##### Syntaxe

VLOOKUP(valeur de recherche; base de données; colonne; méthode de tri)

##### Paramètres

*Commentaire* : Valeur de recherche, *Type* : Chaîne / numérique

*Commentaire* : Base de données, *Type* : Matrice

*Commentaire* : Colonne, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Méthode de tri(optionnel), *Type* : Une valeur de vérité (VRAI ou FAUX)

#### 8.1.10 Mathématiques

##### 8.1.10.1 ABS

La fonction ABS() renvoie la valeur absolue du nombre à virgule flottante x.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

ABS(x)

### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

ABS(12,5) vaut 12,5

### Exemples

ABS(-12,5) vaut 12,5

### 8.1.10.2 CEIL

La fonction CEIL() arrondit x à l'entier le plus proche, plus grand que x et renvoyant cette valeur comme un double.

*Type de retour* : Un entier (comme 0, -5, 14)

### Syntaxe

CEIL(x)

### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

CEIL(12,5) vaut 13

### Exemples

CEIL(-12,5) vaut -12

### Fonctions en relation

CEILING

FLOOR

ROUND

ROUNDUP

### 8.1.10.3 CEILING

La fonction CEILING() arrondit en supérieur x (à partir de zéro) au multiple le plus proche de « Signification », qui est plus grand que x. La valeur par défaut de « Signification » est 1 (ou -1 si la valeur est négative), ce qui correspond à un arrondi vers l'entier le plus proche. Si le paramètre « Méthode » est différent de zéro, la fonction arrondit vers zéro, plutôt que vers l'infini positif.

*Type de retour* : Un entier (comme 0, -5, 14)

### Syntaxe

CEILING(x)

### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Signification (optionnel), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Méthode (optionnel), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

CEILING(12,5) vaut 13

### Exemples

CEILING(6,43;4) vaut 8

### Exemples

CEILING(-6.43; -4; 1) vaut -8

**Exemples**

CEILING(-6.43; -4; 0) vaut -4

**Fonctions en relation**

CEIL  
FLOOR  
ROUND  
ROUNDUP

**8.1.10.4 COUNT**

Cette fonction renvoie le nombre d'arguments entiers ou de flottants passés. Vous pouvez compter en utilisant une zone : COUNT(A1 :B5) ou en utilisant une liste de valeurs comme COUNT(12;5;12.5).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

COUNT(valeur;valeur;valeur...)

**Paramètres**

*Commentaire* : Valeurs, *Type* : FLOAT

**Exemples**

COUNT(-5;"KSpread";2) renvoie 2

**Exemples**

COUNT(5) renvoie 1

**Fonctions en relation**

COUNTA  
COUNTIF  
SUM

**8.1.10.5 COUNTA**

Cette fonction renvoie le nombre d'arguments non vides passés. Vous pouvez compter en utilisant une zone : COUNTA(A1 :B5) ou en utilisant une liste de valeurs comme COUNTA(12;5;12,5).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

COUNTA(valeur;valeur;valeur...)

**Paramètres**

*Commentaire* : Valeurs, *Type* : FLOAT

**Exemples**

COUNTA(-5;"KSpread";2) renvoie 3

**Exemples**

COUNTA(5) renvoie 1

**Fonctions en relation**

COUNT  
COUNTIF

#### 8.1.10.6 COUNTBLANK

Cette fonction renvoie le compte de toutes les cellules vides dans la plage.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

COUNTBLANK(plage)

##### Paramètres

*Commentaire* : Plage de cellules, *Type* : Échelle

##### Exemples

COUNTBLANK(A1 :B5)

##### Fonctions en relation

COUNT  
COUNTA  
COUNTIF

#### 8.1.10.7 COUNTIF

La fonction COUNTIF() renvoie le nombre de cellules dans la plage donnée en paramètre.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

COUNTIF(plage;critère)

##### Paramètres

*Commentaire* : Échelle, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Critères, *Type* : Texte

##### Exemples

COUNTIF(A2 :A3;"14") renvoie 1 si A1 vaut -4 et A2 vaut 14

##### Fonctions en relation

COUNT  
SUMIF

#### 8.1.10.8 CUR

La fonction CUR() renvoie la racine cubique positive de x.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

CUR(x)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

CUR(27) vaut 3

##### Fonctions en relation

SQRT

#### 8.1.10.9 DIV

La fonction DIV() divise la première valeur par chacune des valeurs suivantes, successivement.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

DIV(valeur;valeur;...)

##### Paramètres

*Commentaire* : Valeurs, *Type* : FLOAT

##### Exemples

DIV(20;2;2) renvoie 5

##### Exemples

DIV(25;2.5) renvoie 10

##### Fonctions en relation

[MULTIPLY](#)

[MOD](#)

#### 8.1.10.10 EPS

EPS() renvoie l'épsilon machine. C'est la différence entre 1 et le plus grand nombre à virgule flottante. Du fait que les ordinateurs utilisent un nombre fini de chiffres, les erreurs d'arrondi sont inhérentes (mais habituellement insignifiantes) à tous les calculs.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

EPS()

##### Paramètres

##### Exemples

Sur la plupart des systèmes, ceci renvoie  $2^{-52}=2,2204460492503131e-16$

##### Exemples

$0,5*EPS()$  renvoie « l'arrondi à l'unité ». Cette valeur est intéressante du fait qu'elle est le plus grand nombre  $x$  où  $(1+x)^{-1}=0$  (dû aux erreurs d'arrondi).

##### Exemples

EPS() est si petit que Calligra Sheets affiche  $1+eps()$  comme 1

##### Exemples

Prenez un nombre  $x$  entre 0 et EPS(). Observez que  $1+x$  arrondit  $x$  soit à 0, soit à EPS() en utilisant l'équation  $(1+x)^{-1}$

#### 8.1.10.11 EVEN

La fonction EVEN() renvoie le nombre arrondi à l'entier supérieur pair le plus proche.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

EVEN(valeur)

##### Paramètres

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

EVEN(1,2) renvoie 2

##### Exemples

EVEN(2) renvoie 2

##### Fonctions en relation

[ODD](#)

#### 8.1.10.12 EXP

La fonction EXP() renvoie la valeur de e (la base des logarithmes népériens), élevée à la puissance x.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

EXP(x)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

EXP(9) vaut 8 103,083 927 58

##### Exemples

EXP(-9) vaut 0,000 123 41

##### Fonctions en relation

[LN](#)

#### 8.1.10.13 FACT

La fonction FACT() calcule la factorielle du paramètre. L'expression mathématique est (valeur)!.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

FACT(nombre)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

FACT(10) renvoie 3 628 800

##### Exemples

FACT(0) renvoie 1

#### 8.1.10.14 FACTDOUBLE

La fonction FACTDOUBLE() calcule la factorielle double d'un nombre, c'est-à-dire x!!.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

FACTDOUBLE(nombre)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

FACTDOUBLE(6) renvoie 48

##### Exemples

FACTDOUBLE(7) renvoie 105

#### 8.1.10.15 FIB

La fonction FIB calcule le *n*ème terme d'une séquence de Fibonacci (1, 1, 2, 3, 5, 8, 13, 21...), dans laquelle chaque nombre, après les deux premiers, est la somme des deux nombres immédiatement précédents. @FIB(0) est défini pour être 0.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

FIB(*n*)

##### Paramètres

*Commentaire* : Nième terme, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

FIB(9) renvoie 34

##### Exemples

FIB(26) renvoie 121393

#### 8.1.10.16 FLOOR

Arrondir un nombre *x* à l'entier inférieur vers le multiple le plus proche du second paramètre significatif.

La fonction FLOOR() arrondit « *x* » à la valeur inférieure (vers zéro) vers le plus proche multiple de « Signification », qui est plus petite que *x*. La valeur « Signification » par défaut est 1, si « *x* » est positif. Elle est de -1, si la valeur est négative, ce qui signifie un arrondi vers l'entier le plus proche. Si le mode est donné et est différent de zéro, le montant de « *x* » est arrondi de zéro vers un multiple de « Signification » et ensuite le signe est appliqué. sinon, il est arrondi vers l'infini négatif. Si aucun des deux paramètres « *x* » ou « Signification » est zéro, le résultat est zéro.

*Type de retour* : Un entier (comme 0, -5, 14)

##### Syntaxe

FLOOR(*x*)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Signification (optionnel), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Méthode (optionnel), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

FLOOR(12,5) vaut 12

##### Exemples

FLOOR(-12,5) vaut -13

##### Exemples

FLOOR(5; 2) vaut 4

##### Exemples

FLOOR(5; 2.2) vaut 4.4

##### Fonctions en relation

CEIL

CEILING

ROUND

ROUNDDOWN

#### 8.1.10.17 GAMMA

La fonction GAMMA() renvoie la valeur de la fonction gamma.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

GAMMA(valeur)

##### Paramètres

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

GAMMA(1) renvoie 1

##### Fonctions en relation

[FACT](#)

#### 8.1.10.18 GCD

La fonction GCD() renvoie le plus grand dénominateur commun pour deux valeurs entières ou plus.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

GCD(valeur;valeur)

##### Paramètres

*Commentaire* : Premier nombre, *Type* : Une plage de nombres entiers (comme 1, 132, 2344)

*Commentaire* : Deuxième nombre, *Type* : Une plage de nombres entiers (comme 1, 132, 2344)

*Commentaire* : Troisième nombre, *Type* : Une plage de nombres entiers (comme 1, 132, 2344)

##### Exemples

GCD(6;4) renvoie 2

##### Exemples

GCD(10;20) renvoie 10

##### Exemples

GCD(20;15;10) renvoie 5

##### Fonctions en relation

[LCM](#)

#### 8.1.10.19 G\_PRODUCT

La fonction G\_PRODUCT() est identique à KPRODUCT. Elle est fournie pour la compatibilité avec Gnumeric.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

G\_PRODUCT(valeur;valeur;...)

##### Paramètres

*Commentaire* : Valeurs, *Type* : FLOAT

##### Fonctions en relation

[KPRODUCT](#)



#### 8.1.10.20 INT

La fonction INT() renvoie la partie entière de la valeur.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

INT(x)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

INT(12,55) vaut 12

##### Exemples

INT(15) vaut 15

##### Fonctions en relation

FLOOR  
QUOTIENT

#### 8.1.10.21 INV

Cette fonction multiplie chaque valeur par -1.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

INV(Valeur)

##### Paramètres

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

INV(-5) vaut 5

##### Exemples

INV(5) vaut -5

##### Exemples

INV(0) vaut 0

#### 8.1.10.22 KPRODUCT

La fonction KPRODUCT() calcule le produit de toutes les valeurs données en paramètres. Vous pouvez calculer le produit d'une zone : KPRODUCT(A1 :B5) ou d'une liste de valeurs comme KPRODUCT(12;5;12,5). Si aucune valeur numérique n'est trouvée, il est retourné 1.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

KPRODUCT(valeur;valeur;...)

##### Paramètres

*Commentaire* : Valeurs, *Type* : FLOAT

##### Exemples

KPRODUCT(3;5;7) vaut 105

##### Exemples

KPRODUCT(12,5;2) vaut 25

##### Fonctions en relation

G\_PRODUCT  
MULTIPLY  
PRODUCT

#### 8.1.10.23 LCM

La fonction LCM() renvoie le plus petit commun multiple pour deux valeurs à virgule flottante ou plus

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

LCM(valeur ;valeur)

##### Paramètres

*Commentaire* : Premier nombre, *Type* : FLOAT

*Commentaire* : Deuxième nombre, *Type* : FLOAT

##### Exemples

LCM(6;4) renvoie 12

##### Exemples

LCM(1,5;2,25) renvoie 4,5

##### Exemples

LCM(2;3;4) renvoie 12

##### Fonctions en relation

[GCD](#)

#### 8.1.10.24 LN

La fonction LN() renvoie le logarithme népérien de x.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

LN(x)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

LN(0,8) vaut -0,22314355

##### Exemples

LN(0) vaut -inf

##### Fonctions en relation

[LOG](#)

[LOG10](#)

[LOG2](#)

#### 8.1.10.25 LOG

La fonction LOG() renvoie le log en base 10 de x.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

LOG(x)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante, supérieure à zéro, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Exemples**

LOG(0,8) vaut -0,09691001

**Exemples**

LOG(0) est une erreur

**Fonctions en relation**

LN  
LOGN  
LOG10  
LOG2

**8.1.10.26 LOG10**

La fonction LOG10() renvoie le logarithme en base 10 de l'argument

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

LOG10(x)

**Paramètres**

*Commentaire* : Une valeur positive à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Exemples**

LOG10(10) vaut 1

**Exemples**

LOG10(0) est une erreur

**Fonctions en relation**

LN  
LOGN  
LOG  
LOG2

**8.1.10.27 LOG2**

La fonction LOG2() renvoie le log en base 2 de x.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

LOG2(x)

**Paramètres**

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Exemples**

LOG2(0,8) vaut -0,32192809

**Exemples**

LOG2(0) vaut -inf.

**Fonctions en relation**

LN  
LOGN  
LOG  
LOG10

### 8.1.10.28 LOGN

La fonction LOGn() renvoie le log en base n de x.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

LOGn(valeur;base)

#### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Base, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

LOGn(12;10) vaut 1,07918125

#### Exemples

LOGn(12;2) vaut 3,5849625

#### Fonctions en relation

[LOG](#)

[LN](#)

[LOG10](#)

[LOG2](#)

### 8.1.10.29 MAX

La fonction MAX() renvoie la plus grande des valeurs données en paramètres. Les chaînes et les valeurs logiques sont ignorées.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

MAX(valeur;valeur;...)

#### Paramètres

*Commentaire* : Valeurs, *Type* : FLOAT

#### Exemples

MAX(12;5; 7) renvoie 12

#### Exemples

MAX(12,5; 2) renvoie 12,5

#### Exemples

MAX(0,5; 0,4; Vrai; 0,2) renvoie 0,5

#### Fonctions en relation

[COUNT](#)

[COUNTA](#)

[MAXA](#)

[MIN](#)

[MINA](#)

### 8.1.10.30 MAXA

La fonction MAXA() renvoie la plus grande des valeurs données en paramètres sans ignorer les valeurs booléennes. "Vrai" vaut 1 et "Faux" vaut 0. Les chaînes sont ignorées.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

MAXA(valeur;valeur;...)

**Paramètres**

*Commentaire* : Valeurs, *Type* : FLOAT

**Exemples**

MAXA(12;5;7) renvoie 12

**Exemples**

MAXA(12,5;2) renvoie 12,5

**Exemples**

MAXA(0,5;0,4;Vrai.0,2) renvoie 1

**Fonctions en relation**

COUNT

COUNTA

MAX

MIN

MINA

### 8.1.10.31 MDETERM

La fonction MDETERM renvoie le déterminant d'une matrice donnée. Celle-ci doit être de la forme n x n.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

MDETERM(matrice)

**Paramètres**

*Commentaire* : Échelle, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Exemples**

MDETERM(A1 :C3)

**Fonctions en relation**

MMULT

### 8.1.10.32 Min

La fonction MIN() renvoie la plus petite des valeurs données en paramètres. Les chaînes et les valeurs logiques sont ignorées.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

MIN(valeur;valeur;...)

**Paramètres**

*Commentaire* : Valeurs, *Type* : FLOAT

**Exemples**

MIN(12;5; 7) renvoie 5

**Exemples**

MIN(12,5; 2) renvoie 2

**Exemples**

MIN(0,4; 2; Faux; 0,7) renvoie 0,4

**Fonctions en relation**

COUNT

COUNTA

MAX

MAXA

MINA

### 8.1.10.33 MINA

La fonction MINA() renvoie la plus petite des valeurs données en paramètres sans ignorer les valeurs booléennes. "Vrai" vaut 1 et "Faux" vaut 0. Les chaînes sont ignorées.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

MINA(valeur ;valeur ;...)

#### Paramètres

*Commentaire* : Valeurs, *Type* : FLOAT

#### Exemples

MINA(12;5;7) renvoie 5

#### Exemples

MINA(12,5;2) renvoie 2

#### Exemples

MINA(0,4; 2; Faux; 0,7) renvoie 0

#### Fonctions en relation

COUNT  
COUNTA  
MAX  
MAXA  
MIN

### 8.1.10.34 MINVERSE

Calcule l'inverse de la matrice

La matrice multipliée avec son résultat inverse dans la matrice d'unité de la même dimension.

Les matrices inversables ont un déterminant non nul.

*Type de retour* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

MINVERSE(matrice)

#### Paramètres

*Commentaire* : Matrice, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

MINVERSE(A1 :C3)

#### Fonctions en relation

MDETERM

### 8.1.10.35 MMULT

La fonction MMULT multiplie deux matrices. Le nombre de colonnes de la première doit être égal au nombre de lignes de la seconde. Le résultat est une matrice.

*Type de retour* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

MMULT(matrice1 ;matrice2)

### Paramètres

*Commentaire* : Première matrice, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Seconde matrice, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

MMULT(A1 :C3)

### Fonctions en relation

[MDETERM](#)

#### 8.1.10.36 MOD

La fonction MOD() renvoie le reste après une division. Si le deuxième paramètre est nul, la fonction renvoie #DIV/0.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

### Syntaxe

MOD(valeur;valeur)

### Paramètres

*Commentaire* : Valeur à virgule flottante, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Valeur à virgule flottante, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

MOD(12;5) renvoie 2

### Exemples

MOD(5;5) renvoie 0

### Fonctions en relation

[DIV](#)

#### 8.1.10.37 MROUND

La fonction MROUND() renvoie la valeur arrondie au multiple spécifié. La valeur et le multiple doivent avoir le même signe.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

MROUND(valeur;multiple)

### Paramètres

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Multiple, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

MROUND(1,252;0,5) vaut 1,5

### Exemples

MROUND(-1,252;-0,5) vaut -1,5

### Fonctions en relation

[ROUND](#)

#### 8.1.10.38 MULTINOMIAL

La fonction MULTINOMIAL() renvoie le multinomial de chaque nombre donné en paramètre. On utilise cette formule pour MULTINOMIAL(a,b,c) :

$(a+b+c)! / a!b!c!$

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

MULTINOMIAL(valeur;valeur;...)

##### Paramètres

*Commentaire* : Valeurs, *Type* : FLOAT

##### Exemples

MULTINOMIAL(3;4;5) vaut 27 720

#### 8.1.10.39 MULTIPLY

La fonction MULTIPLY() multiplie toutes les valeurs données en paramètres Vous pouvez multiplier les valeurs d'une zone MULTIPLY(A1 :B5) ou une liste de valeurs comme MULTIPLY(12;5;12,5). Elle est équivalente à PRODUCT.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

MULTIPLY(valeur;valeur;...)

##### Paramètres

*Commentaire* : Valeurs, *Type* : FLOAT

##### Exemples

MULTIPLY(12;5;7) vaut 420

##### Exemples

MULTIPLY(12,5;2) vaut 25

##### Fonctions en relation

DIV  
PRODUCT  
KPRODUCT

#### 8.1.10.40 MUNIT

Crée la matrice d'unité d'une dimension donnée.

*Type de retour* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

MUNIT(dimension)

##### Paramètres

*Commentaire* : Dimension, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

MUNIT(3) crée une matrice d'unité de 3x3

##### Fonctions en relation

MINVERSE



#### 8.1.10.41 ODD

La fonction ODD() renvoie le nombre arrondi à l'entier impair supérieur (ou inférieur pour des valeurs négatives) le plus proche. Par définition, ODD(0) est 1.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

ODD(valeur)

##### Paramètres

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

ODD(1,2) renvoie 3

##### Exemples

ODD(2) renvoie 3

##### Exemples

ODD(2) renvoie -3

##### Fonctions en relation

[EVEN](#)

#### 8.1.10.42 POW

La fonction POW(x,y) renvoie la valeur de x élevé à la puissance y. Elle agit comme POWER.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

POW(valeur;valeur)

##### Paramètres

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

POW(1,2;3,4) vaut 1,8572

##### Exemples

POW(2;3) vaut 8

##### Fonctions en relation

[POWER](#)

#### 8.1.10.43 POWER

La fonction POWER(x,y) renvoie la valeur de x élevé à la puissance y.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

POWER(valeur;valeur)

##### Paramètres

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Exemples**

POWER(1,2;3,4) vaut 1,8572

**Exemples**

POWER(2;3) vaut 8

**Fonctions en relation**

[POW](#)

**8.1.10.44 PRODUCT**

La fonction PRODUCT() calcule le produit de toutes les valeurs données en paramètres. Vous pouvez calculer le produit d'une zone : PRODUCT(A1 :B5) ou d'une liste de valeurs comme PRODUCT(12;5;12,5). Si aucune valeur numérique n'est trouvée, il est retourné 0.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

PRODUCT(valeur;valeur;...)

**Paramètres**

*Commentaire* : Valeurs, *Type* : FLOAT

**Exemples**

PRODUCT(3;5;7) vaut 105

**Exemples**

PRODUCT(12.5;2) vaut 25

**Fonctions en relation**

[MULTIPLY](#)

[KPRODUCT](#)

**8.1.10.45 QUOTIENT**

La fonction QUOTIENT renvoie la partie entière de numérateur / dénominateur.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

**Syntaxe**

QUOTIENT(numérateur;dénominateur)

**Paramètres**

*Commentaire* : Numérateur, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Dénominateur, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Exemples**

QUOTIENT(21;4) renvoie 5

**Fonctions en relation**

[INT](#)

**8.1.10.46 RAND**

La fonction RAND() renvoie un nombre pseudo-aléatoire entre 0 et 1.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

RAND()

**Paramètres**

**Exemples**

RAND() vaut par exemple 0,783 099 22...

**Fonctions en relation**

[RANDBETWEEN](#)

[RANDEXP](#)

#### 8.1.10.47 RANDBERNOULLI

La fonction RANDBERNOULLI() renvoie un nombre pseudo-aléatoire à distribution de Bernoulli.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

RANDBERNOULLI(x)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante (entre 0 et 1), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

RANDBERNOULLI(0,45)

##### Fonctions en relation

[RAND](#)

#### 8.1.10.48 RANDBETWEEN

La fonction RANDBETWEEN() renvoie un nombre pseudo-aléatoire entre une valeur-plancher et une valeur-plafond. Si plancher > plafond, cette fonction renvoie Err.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

RANDBETWEEN(plancher ;plafond)

##### Paramètres

*Commentaire* : Valeur plancher, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Valeur plafond, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

RANDBETWEEN(12;78) vaut par exemple 61,081 1...

##### Fonctions en relation

[RAND](#)

#### 8.1.10.49 RANDBINOM

La fonction RANDBINOM() renvoie un nombre pseudo-aléatoire à distribution binomiale.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

RANDBINOM(x)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante (entre 0 et 1), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Tentatives (supérieur à 0), *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

RANDBINOM(4)

##### Fonctions en relation

[RAND](#)

[RANDNEGBINOM](#)

#### 8.1.10.50 RANDEXP

La fonction RANDEXP() renvoie un nombre pseudo-aléatoire distribué exponentiellement.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

RANDEXP(x)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante (supérieure à 0), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

RANDEXP(0,88)

##### Fonctions en relation

[RAND](#)

#### 8.1.10.51 RANDNEGBINOM

La fonction RANDNEGBINOM() renvoie un nombre négatif pseudo-aléatoire à distribution binomiale.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

RANDNEGBINOM(x)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante (entre 0 et 1), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Échecs (supérieur à 0), *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

RANDNEGBINOM(4)

##### Fonctions en relation

[RAND](#)

[RANDBINOM](#)

#### 8.1.10.52 RANDNORM

La fonction RANDNORM() renvoie un nombre pseudo-aléatoire à distribution normale (Gaussien).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

RANDNORM(mu ; sigma)

##### Paramètres

*Commentaire* : Valeur moyenne de la distribution normale, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Écart type de la distribution normale, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

RANDNORM(0 ; 1)

##### Fonctions en relation

[RAND](#)

#### 8.1.10.53 RANDPOISSON

La fonction RANDPOISSON() renvoie un nombre pseudo-aléatoire selon une distribution de Poisson.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

RANDPOISSON(x)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante (supérieure à 0), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

RANDPOISSON(4)

##### Fonctions en relation

[RAND](#)

#### 8.1.10.54 ROOTN

La fonction ROOTN() renvoie la racine nième positive de x.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

ROOTN(x;n)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

ROOTN(9;2) vaut 3

##### Fonctions en relation

[SQRT](#)

#### 8.1.10.55 ROUND

La fonction ROUND(valeur;[chiffres]) renvoie la valeur arrondie. "Chiffres" est le nombre de chiffres auquel vous voulez arrondir ce nombre. Si "Chiffres" est 0 ou omis, la valeur est arrondie à l'entier le plus proche. Si "Chiffres" est plus petit que zéro, la partie entière correspondante du nombre est arrondie.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

ROUND(valeur;[chiffres])

##### Paramètres

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Digitales, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

ROUND(1,252;2) vaut 1,25

##### Exemples

ROUND(-1,252;2) vaut -1,25

**Exemples**

ROUND(1,252;2) vaut 1,26

**Exemples**

ROUND(-12,25;-1) vaut -10

**Exemples**

ROUND(-1,252;0) vaut -1

**Fonctions en relation**

[MROUND](#)

[ROUND](#)

[ROUNDUP](#)

### 8.1.10.56 ROUNDDOWN

La fonction ROUNDDOWN(valeur ;[chiffres]) renvoie l'arrondi inférieur de la valeur la plus petite. "Chiffres" est le nombre de chiffres auquel vous voulez arrondir ce nombre. Si "chiffres" vaut 0 ou est omis, la valeur est arrondie à l'entier le plus proche.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

ROUNDDOWN(valeur ;[chiffres])

**Paramètres**

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Digitales, *Type* : Un nombre entier (comme 1, 132, 2344)

**Exemples**

ROUNDDOWN(1,252) vaut 1

**Exemples**

ROUNDDOWN(1,252;2) vaut 1,25

**Exemples**

ROUNDDOWN(-1,252;2) vaut -1,26

**Exemples**

ROUNDDOWN(-1,252) vaut -1

**Fonctions en relation**

[ROUND](#)

[ROUNDUP](#)

### 8.1.10.57 ROUNDUP

La fonction ROUNDUP(valeur ;[chiffres]) renvoie l'arrondi supérieur de la valeur absolue la plus grande. "Chiffres" est le nombre de chiffres auquel vous voulez arrondir ce nombre. Si "chiffres" vaut 0 ou est omis, la valeur est arrondie à l'entier le plus proche.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

ROUNDUP(valeur ;[chiffres])

**Paramètres**

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Digitales, *Type* : Un nombre entier (comme 1, 132, 2344)

**Exemples**

ROUNDUP(1,252) vaut 2

**Exemples**

ROUNDUP(1,252;2) vaut 1,26

**Exemples**

ROUNDUP(-1,252;2) vaut -1,26

**Exemples**

ROUNDUP(-1,252) vaut -2

**Fonctions en relation**

[ROUND](#)

[ROUNDDOWN](#)

### 8.1.10.58 SERIESSUM

La fonction SERIESSUM() renvoie la somme d'une série de puissances.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

SERIESSUM( X; N; M; coefficients)

**Paramètres**

*Commentaire* : X la variable indépendante des séries de puissances, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : N la puissance initiale à laquelle X doit être élevé, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : M l'incrément avec lequel augmenter N pour chaque terme dans les séries, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Coefficients : une série de coefficients par lesquels chaque puissance successive de la variable X est multiplié, *Type* : FLOAT

**Exemples**

SERIESSUM(2;0;2;{1;2}) renvoie 9

### 8.1.10.59 SIGN

Cette fonction renvoie -1 si le nombre est négatif, 0 si le nombre est nul et 1 si le nombre est positif.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

**Syntaxe**

SIGN(Valeur)

**Paramètres**

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Exemples**

SIGN(5) vaut 1

**Exemples**

SIGN(0) vaut 0

**Exemples**

SIGN(-5) vaut -1

#### 8.1.10.60 SQRT

La fonction SQRT() renvoie la racine carrée positive de l'argument x. Elle indique une erreur si l'argument est négatif.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SQRT(x)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

SQRT(9) vaut 3

##### Exemples

SQRT(-9) est une erreur

##### Fonctions en relation

[IMSQRT](#)

#### 8.1.10.61 SQRTPI

La fonction SQRTPI() renvoie la racine carrée positive de x\*Pi. Elle indique une erreur si l'argument est négatif.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SQRTPI(x)

##### Paramètres

*Commentaire* : Une valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

SQRTPI(2) vaut 2,506628

#### 8.1.10.62 SUBTOTAL

La fonction SUBTOTAL() renvoie un sous-total d'une liste donnée d'arguments ignorant les autres résultats de sous-total. La fonction peut être l'un des nombres suivants : 1 - Average, 2 - Count, 3 - CountA, 4 - Max., 5 - Min 6 - Product, 7 - StDev, 8 - StDevP, 9 - Sum, 10 - Var, 11 - VarP.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SUBTOTAL(fonction;valeur)

##### Paramètres

*Commentaire* : Fonction, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Valeurs, *Type* : FLOAT

##### Exemples

Si A1 :A5 contient 7, 24, 23, 56 et 9 :

##### Exemples

SUBTOTAL(1; A1 :A5) renvoie 23,8

##### Exemples

SUBTOTAL(4; A1 :A5) renvoie 56



**Exemples**

SUBTOTAL(9; A1 :A5) renvoie 119

**Exemples**

SUBTOTAL(11; A1 :A5) renvoie 307,76

**Fonctions en relation**

AVERAGE  
COUNT  
COUNTA  
MAX  
MIN  
PRODUCT  
STDEV  
STDEVP  
SUM  
VAR  
VARP

**8.1.10.63 SUM**

La fonction SUM() calcule la somme de toutes les valeurs données en paramètres. Vous pouvez calculer la somme d'une zone (SUM(A1 :B5)) ou une liste de valeurs (comme SUM(12;5;12,5)).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

SUM(valeur,valeur;...)

**Paramètres**

*Commentaire* : Valeurs, *Type* : FLOAT

**Exemples**

SUM(12;5;7) vaut 24

**Exemples**

SUM(12,5;2) vaut 14,5

**Fonctions en relation**

SUMA  
SUMSQ  
SUMIF

**8.1.10.64 SUMA**

La fonction SUMA() calcule la somme de toutes les valeurs données en paramètres. Vous pouvez calculer la somme d'une zone (SUMA(A1 :B5)) ou une liste de valeurs (comme SUMA(12;5;12,5)). Si un paramètre contient du texte ou la valeur booléenne Faux, il compte comme 0. Si un paramètre vaut Vrai, il compte pour 1.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

SUM(valeur,valeur;...)

**Paramètres**

*Commentaire* : Valeurs, *Type* : FLOAT

**Exemples**

SUMA(12;5;7) vaut 24

**Exemples**

SUMA(12,5;2;Vrai) vaut 15,5

**Fonctions en relation**

SUM  
SUMSQ

#### 8.1.10.65 SUMIF

La fonction SUMIF() calcule la somme de toutes les valeurs données en paramètre si elles correspondent au critère de vérification. Le paramètre de plage d'addition est optionnel. Si celui-ci n'est pas fourni, les valeurs dans la plage de vérification sont additionnées. La longueur de la plage de vérification doit être inférieure ou égale à la longueur de la plage d'addition.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SUMIF(rang\_vérification; critère; rang\_addition)

##### Paramètres

*Commentaire* : Plage de vérification, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Critères, *Type* : Texte

*Commentaire* : Plage d'addition, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

SUMIF(A1 :A4; ">1") fait la somme de toutes les valeurs dans la plage A1 :A4 qui sont supérieures à 1

##### Exemples

SUMIF(A1 :A4; "=0"; B1 :B4) fait la somme de toutes les valeurs dans la plage B1 :B4 si les valeurs correspondantes dans la plage A1 :A4 sont égales à 0

##### Fonctions en relation

SUM  
COUNTIF

#### 8.1.10.66 SUMSQ

La fonction SUMSQ() calcule la somme des carrés de toutes les valeurs données en paramètres. Vous pouvez calculer la somme d'une zone (SUMSQ(A1 :B5)) ou une liste de valeurs (comme SUMSQ(12;5;12,5)).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SUMSQ(valeur,valeur;...)

##### Paramètres

*Commentaire* : Valeurs, *Type* : FLOAT

##### Exemples

SUMSQ(12;5;7) vaut 218

##### Exemples

SUMSQ(12,5;2) vaut 173

##### Fonctions en relation

SUM

#### 8.1.10.67 TRANSPOSE

renvoie la transposition d'une matrice, cela signifie que les lignes et colonne de la matrice sont inchangées.

*Type de retour* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

TRANSPOSE(matrice)

### Paramètres

*Commentaire* : Matrice, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

TRANSPOSE(A1 :C3)

### 8.1.10.68 TRUNC

La fonction TRUNC() tronque une valeur numérique à une certaine position. Si la précision est omise, elle est supposée valoir 0.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

TRUNC(valeur, précision)

### Paramètres

*Commentaire* : Valeur à virgule flottante, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Précision, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

TRUNC(1,2) renvoie 1

### Exemples

TRUNC(213,232 ; 2) renvoie 213,23

### Fonctions en relation

[ROUND](#)

[ROUNDDOWN](#)

[ROUNDUP](#)

## 8.1.11 Statistiques

### 8.1.11.1 AVEDEV

La fonction AVEDEV() calcule la moyenne des déviations absolues d'un lot de données par rapport à leur moyenne.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

AVEDEV(valeur ; valeur ; ...)

### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

AVEDEV(11,4;17,3;21,3;25,9;40,1) renvoie 7.84

### Exemples

AVEDEV(A1 :A5) ...

### 8.1.11.2 AVERAGE

La fonction AVERAGE() calcule la moyenne de toutes les valeurs données en paramètres. Vous pouvez calculer la moyenne d'une zone (AVERAGE(A1 :B5)) ou d'une liste de valeurs (comme AVERAGE(12 ;5 ;12,5)).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

AVERAGE(valeur ;valeur ;...)

#### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

AVERAGE(12 ;5 ;7) vaut 8

#### Exemples

AVERAGE(12,5 ;2) vaut 7,25

### 8.1.11.3 AVERAGEA

La fonction AVERAGEA() calcule la moyenne des arguments donnés. Les nombres, textes et valeurs logiques sont inclus dans le calcul. Si la cellule contient du texte ou des arguments évalués à Faux, ils sont comptés comme une valeur de zéro (0). Si l'argument s'évalue à Vrai, il est compté comme un (1). Notez que les cellules vides ne sont pas comptées.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

AVERAGEA(valeur ;valeur ;...)

#### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs des chaînes, *Type* : Texte

#### Exemples

AVERAGEA(11,4 ;17,3 ;"dutexte" ;25,9 ;40,1) vaut 18,94

### 8.1.11.4 BETADIST

La fonction BETADIST() renvoie la fonction de densité de probabilité cumulative beta.

Les troisième et quatrième paramètres sont optionnels. Ils précisent les limites basse et haute, sinon, les valeurs par défaut sont respectivement 0,0 et 1,0.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

BETADIST(nombre ;alpha ;beta ;début ;fin ;[cumul=VRAI])

### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Paramètres alpha, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Paramètres beta, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Début, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : End, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Cumul, *Type* : Une valeur de vérité (VRAI ou FAUX)

### Exemples

BETADIST(0,2859;0,2606;0,8105) vaut 0,675 444

### Exemples

BETADIST(0,2859;0,2606;0,8105;0,2;0,9) vaut 0,537 856

### 8.1.11.5 BETAINV

La fonction BETAINV() renvoie l'inverse de BETADIST(x;alpha;beta;a;b;TRUE()).

Les paramètres de début et de fin sont optionnels. Ils précisent les limites basse et haute, sinon, les valeurs par défaut sont respectivement 0,0 et 1,0.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

BETAINV(nombre;alpha;beta [; début=0 [; fin=1]])

### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Paramètres alpha, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Paramètres beta, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Début, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : End, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

BETADIST(BETAINV(0.1;3;4);3;4) vaut 0.1

### Exemples

BETADIST(BETAINV(0.3;3;4);3;4) égale 0.3

### 8.1.11.6 BINO

La fonction BINO() renvoie la distribution binomiale.

Le premier paramètre est le nombre d'essais, le deuxième paramètre est le nombre de succès et le troisième est la probabilité de succès. Le nombre d'essais doit être plus grand que le nombre de succès et la probabilité doit être inférieure à 1.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

BINO(essais;succès;prob\_de\_succès)

### Paramètres

*Commentaire* : Nombre d'essais, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombre de succès, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Probabilité de succès, *Type* : Double

### Exemples

BINO(12;9;0,8) renvoie 0,236 223 201

#### 8.1.11.7 CHIDIST

La fonction CHIDIST() renvoie la valeur de probabilité, indiquée par le Chi2, qu'une hypothèse soit confirmée.

CHIDIST compare la valeur du Chi2 à donner pour un échantillon aléatoire qui est calculé de la somme de (valeur observée-valeur attendue)<sup>2</sup> / valeur attendue pour toutes les valeurs avec la distribution CHI2 théorique et détermine à partir de ceci la probabilité d'erreur pour l'hypothèse à tester.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

CHIDIST(nombre;degrés\_liberté)

##### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Degrés de liberté, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

CHIDIST(13,27;5) renvoie 0,021

#### 8.1.11.8 COMBIN

La fonction COMBIN() calcule un nombre de combinaisons. Le premier paramètre est le nombre total d'éléments. Le deuxième est le nombre d'éléments à choisir. Ces deux éléments doivent être positifs, et le premier paramètre ne doit pas être plus petit que le second, sinon la fonction renvoie une erreur.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

COMBIN(total;choisis)

##### Paramètres

*Commentaire* : Nombre total d'éléments, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombre d'éléments à choisir, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

COMBIN(12;5) renvoie 792

##### Exemples

COMBIN(5;5) renvoie 1

#### 8.1.11.9 COMBINA

La fonction COMBINA() calcule le nombre de combinaisons possibles. Le premier paramètre est le nombre total d'éléments. Le deuxième est le nombre d'éléments à choisir. Ces deux paramètres doivent être positifs, et le premier paramètre ne doit pas être plus petit que le second, sinon la fonction renvoie une erreur.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

COMBIN(total;choisis)

##### Paramètres

*Commentaire* : Nombre total d'éléments, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombre d'éléments à choisir, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

COMBIN(12;5) renvoie 792

##### Exemples

COMBIN(5;5) renvoie 1

#### 8.1.11.10 CONFIDENCE

La fonction CONFIDENCE() renvoie l'intervalle de confiance pour une population moyenne.

Le paramètre alpha doit se situer entre 0 et 1 (non inclus), "stddev" doit être positif et "taille" doit être supérieur ou égal à 1.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

CONFIDENCE(alpha;stddev;taille)

##### Paramètres

*Commentaire* : Niveau de l'intervalle de confiance, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Écart type pour la population totale, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Taille de la population totale, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

CONFIDENCE(0,05;1,5;100) vaut 0,294 059

#### 8.1.11.11 CORREL

La fonction CORREL() calcule le coefficient de corrélation de deux plages de cellules.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

CORREL(plage1 ; plage2)

##### Paramètres

*Commentaire* : Plage de valeurs de cellules, *Type* : Double

*Commentaire* : Deuxième plage de valeurs de cellules, *Type* : Double

##### Exemples

CORREL(A1 :A3 ; B1 :B3)

##### Fonctions en relation

PEARSON

#### 8.1.11.12 COVAR

La fonction COVAR() calcule la covariance de deux plages de cellules.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

COVAR(plage1 ; plage2)

##### Paramètres

*Commentaire* : Plage de valeurs de cellules, *Type* : Double

*Commentaire* : Deuxième plage de valeurs de cellules, *Type* : Double

##### Exemples

COVAR(A1 :A3 ; B1 :B3)

#### 8.1.11.13 DEVSQ

La fonction DEVSQ() calcule la somme des carrés des déviations.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

DEVSQ(valeur;valeur;...)

##### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

##### Exemples

DEVSQ(A1:A5)

##### Exemples

DEVSQ(21; 33; 54; 23) renvoie 684,75

#### 8.1.11.14 EXPONDIST

La fonction EXPONDIST() renvoie la distribution exponentielle.

Le paramètre lambda doit être positif.

Cumulative = 0 calcule la fonction de densité . Cumulative = 1 calcule la distribution.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

EXPONDIST(nombre;lambda;cumul)

##### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Paramètre lambda, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : 0 = densité, 1 = distribution, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

EXPONDIST(3;0,5;0) vaut 0,111 565

##### Exemples

EXPONDIST(3;0,5;1) vaut 0,776 870

#### 8.1.11.15 FDIST

La fonction FDIST() renvoie la distribution f.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

FDIST(nombre;degrés\_liberté\_1;degrés\_liberté\_2)

##### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Degrés de liberté 1, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Degrés de liberté 2, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

FDIST(0,8;8;12) produit 0,61



#### 8.1.11.16 FINV

La fonction FINV() renvoie l'unique nombre positif  $x$  tel que  $FDIST(x;r1;r2) = p$ .

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

FINV(nombre; r1; r2)

##### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Nombre r1, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombre r2, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

FDIST(FINV(0.1;3;4);3;4) vaut 0.1

#### 8.1.11.17 FISHER

Revoit la transformation de Fisher pour  $x$  et crée une fonction proche d'une distribution normale.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

FISHER(nombre)

##### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

FISHER(0,2859) vaut 0,294 096

##### Exemples

FISHER(0,8105) vaut 1,128 485

#### 8.1.11.18 FISHERINV

La fonction FISHERINV() renvoie l'inverse de la transformation de Fisher pour  $x$  et crée une fonction proche d'une distribution normale.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

FISHERINV(nombre)

##### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

FISHERINV(0,2859) vaut 0,278 357

##### Exemples

FISHERINV(0,8105) vaut 0,669 866

#### 8.1.11.19 FREQUENCY

Décompter le nombre de valeurs pour chaque intervalle donné par les valeurs de bordure dans le second paramètre.

La valeur dans le second paramètre détermine la limite supérieure de l'intervalle. L'intervalle inclut la limite supérieure. Le tableau renvoyé est un vecteur de colonne et a plus d'un élément que le second paramètre; le dernier élément représente le nombre de tous les éléments supérieurs à la dernière valeur dans le second paramètre. Si le second paramètre est vide, toutes les valeurs dans le premier paramètre sont comptées.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

FREQUENCY(Plage de données; Plage de binaires)

##### Paramètres

*Commentaire* : Valeurs à virgule flottante qui peuvent être comptées., *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante représentant les limites supérieures des intervalles., *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### 8.1.11.20 GAMMADIST

La fonction GAMMADIST() renvoie la distribution gamma.

Si le dernier paramètre (cumulé) est 0, il calcule la fonction densité; si c'est 1, la distribution est renvoyée.

Les trois premiers paramètres doivent être positifs.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

GAMMADIST(nombre;alpha;beta;cumul)

##### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Paramètres alpha, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Paramètres beta, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Marqueurs cumulés, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

GAMMADIST(0,758;0,1;0,35;1) vaut 0,995 450

##### Exemples

GAMMADIST(0,758;0,1;0,35;0) vaut 0,017 179

#### 8.1.11.21 GAMMAINV

La fonction GAMMAINV() renvoie le nombre unique  $x \geq 0$  tel que  $\text{GAMMAINV}(x;\text{alpha};\text{beta};\text{TRUE}()) = p$ .

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

GAMMAINV(nombre;alpha;beta)

### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Paramètres alpha, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Paramètres beta, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

GAMMADIST(GAMMAINV(0.1;3;4);3;4) vaut 0.1

### Exemples

GAMMADIST(GAMMAINV(0.3;3;4);3;4) vaut 0.3

### 8.1.11.22 GAMMALN

La fonction GAMMA() renvoie le logarithme naturel de la fonction gamma :  $G(x)$ . Le paramètre nombre doit être positif.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

GAMMALN(Nombre)

### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

GAMMALN(2) renvoie 0

### 8.1.11.23 GAUSS

La fonction GAUSS() renvoie la valeur intégrale pour la distribution cumulative normale standard.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

GAUSS(valeur)

### Paramètres

*Commentaire* : Le nombre pour lequel la valeur intégrale de la distribution normale standard doit être calculée, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

GAUSS(0,25) vaut 0,098 706

### 8.1.11.24 GEOMEAN

La fonction GEOMEAN() renvoie la moyenne géométrique des arguments donnés. C'est égal à la racine nième du produit des termes.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

GEOMEAN(valeur; valeur; ...)

### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

### Exemples

GEOMEAN(A1 :A5)

### Exemples

GEOMEAN(21 ; 33 ; 54 ; 23) renvoie 30,458 86

### Fonctions en relation

[HARMEAN](#)

#### 8.1.11.25 HARMEAN

La fonction HARMEAN() calcule la moyenne harmonique des N points de données (N divisé par la somme des inverses des points de données).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

HARMEAN(valeur ;valeur ;...)

### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

### Exemples

HARMEAN(A1 :A5)

### Exemples

HARMEAN(21 ; 33 ; 54 ; 23) renvoie 28,588

### Fonctions en relation

[GEOMEAN](#)

#### 8.1.11.26 HYPGEOMDIST

La fonction HYPGEOMDIST() renvoie la distribution hypergéométrique.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

HYPGEOMDIST(x ; n ; M ; N)

### Paramètres

*Commentaire* : Nombre de succès dans un échantillon, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombre d'essais, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombre total de succès, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Taille de la population, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

HYPGEOMDIST(2 ; 5 ; 6 ; 20) renvoie 0,352 2

#### 8.1.11.27 INTERCEPT

La fonction INTERCEPT() calcule l'interception de la régression linéaire avec l'axe y.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

INTERCEPT(y ;x)

### Paramètres

*Commentaire* : Valeur y (tableau), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur x (tableau), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### 8.1.11.28 INVBINO

La fonction INVBINO() renvoie la distribution binomiale négative. Le premier paramètre est le nombre de tentatives, le deuxième paramètre est le nombre d'échecs et le troisième est la probabilité d'échec. Le nombre d'essais doit être plus grand que le nombre d'échecs et la probabilité doit être inférieure ou égale à 1.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

INVBINO(essais;échecs;prob\_d\_échec)

### Paramètres

*Commentaire* : Nombre d'essais, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombres d'échecs, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Probabilité d'échec, *Type* : Double

### Exemples

INVBINO(12;3;0,2) renvoie 0,236 223 201

#### 8.1.11.29 KURT

La fonction KURT() calcule une estimation sans biais du Kurtosis d'un lot de données. Vous devez fournir au moins 4 valeurs, sinon une erreur est retournée.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

KURT(valeur;valeur;...)

### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

### Exemples

KURT(A1 :A5)

### Exemples

KURT(21 ; 33 ; 54 ; 23) renvoie 1,344 239

### Fonctions en relation

[KURTP](#)

#### 8.1.11.30 KURTP

La fonction KURTP() calcule une population kurtosis d'un lot de données. Vous devez fournir au moins 4 valeurs, sinon, une erreur sera retournée.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

KURTP(valeur;valeur;...)

### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

*Commentaire* : Valeurs à virgule flottante, *Type* : Double

### Exemples

KURTP(A1 :A5)

### Exemples

KURTP(21 ; 33 ; 54 ; 23) renvoie -1,021

### Fonctions en relation

[KURT](#)

#### 8.1.11.31 gros

La fonction LARGE() renvoie la k-ème plus grande valeur d'un lot de données.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

LARGE(plage ; k)

### Paramètres

*Commentaire* : Plage de valeurs de cellules, *Type* : Double

*Commentaire* : Position (depuis le plus grand), *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

A1 : 3, A2 : 1, A3 : 5 => LARGE(A1 :A3 ; 2) renvoie 3

#### 8.1.11.32 LEGACYFDIST

La fonction LEGACYFDIST() renvoie la distribution f.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

LEGACYFDIST(nombre ; degrés\_liberté\_1 ; degrés\_liberté\_2)

### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Degrés de liberté 1, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Degrés de liberté 2, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

LEGACYFDIST(0,8;8;12) produit 0,61

#### 8.1.11.33 LOGINV

La fonction LOGINV() renvoie l'inverse de la distribution cumulative lognormale.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

LOGINV(p ; moyenne ; stdev)

### Paramètres

*Commentaire* : Probabilité, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur moyenne de la distribution logarithmique standard, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Écart type de la distribution logarithmique standard, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

LOGINV(0,1;0;1) vaut 0,277 6

#### 8.1.11.34 LOGNORMDIST

La fonction LOGNORMDIST() renvoie la distribution cumulative lognormale.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

LOGNORMDIST(Nombre;MV;STD)

##### Paramètres

*Commentaire* : Valeur de probabilité pour laquelle la distribution logarithmique standard doit être calculée, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur moyenne de la distribution logarithmique standard, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Écart type de la distribution logarithmique standard, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

LOGNORMDIST(0,1;0;1) vaut 0,01

#### 8.1.11.35 MEDIAN

La fonction MEDIAN() calcule la moyenne de toutes les valeurs données en paramètres. Vous pouvez calculer la moyenne d'une plage (MEDIAN(A1 :B5)) ou d'une liste de valeurs (comme MEDIAN(12;5;12,5)). Les cellules vides sont considérées comme valant zéro, et les cellules contenant du texte seront ignorées.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

MEDIAN(valeur;valeur;...)

##### Paramètres

*Commentaire* : Valeurs à virgule flottante ou plages de valeurs, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante ou plage de valeurs, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante ou plage de valeurs, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante ou plage de valeurs, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante ou plage de valeurs, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

MEDIAN(12; 5; 5.5) vaut 5,5

##### Exemples

MEDIAN(12; 7; 8; 2) vaut 7,5

#### 8.1.11.36 MODE

La fonction MODE() renvoie la valeur la plus fréquente dans le lot de données.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

MODE(nombre; nombre2; ...)

### Paramètres

*Commentaire* : Flottant, *Type* : Double

*Commentaire* : Flottant, *Type* : Double

*Commentaire* : Flottant, *Type* : Double

*Commentaire* : Flottant, *Type* : Double

### Exemples

MODE(12; 14; 12; 15) renvoie 12

### 8.1.11.37 NEGBINOMDIST

La fonction NEGBINOMDIST() renvoie la distribution binomiale négative.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

NEGBINOMDIST(essais ;succès ;prob\_de\_succès)

### Paramètres

*Commentaire* : Nombres d'échecs, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombre de succès, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Probabilité de succès, *Type* : Double

### Exemples

NEGBINOMDIST(2;5;0,55) renvoie 0,152 872 629

### 8.1.11.38 NORMDIST

La fonction NORMDIST() renvoie la distribution cumulative normale.

Nombre est la valeur de la distribution fondée sur laquelle la distribution normale doit être calculée.

MV est le milieu linéaire de la distribution

STD est l'écart-type de la distribution.

K = 0 calcule la fonction de densité; K = 1 calcule la distribution.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

NORMDIST(Nombre;MV;STD;K)

### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Milieu linéaire de la distribution, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Écart-type de la distribution, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : 0 = densité, 1 = distribution, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

NORMDIST(0,859;0,6;0,258;0) vaut 0,934 236

### Exemples

NORMDIST(0,859;0,6;0,258;1) vaut 0,842 281



#### 8.1.11.39 NORMINV

La fonction NORMINV() renvoie l'inverse de la distribution cumulative normale. Le nombre doit se situer entre 0 et 1 (non inclus) et "STD" doit être positif.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

NORMINV(nombre;MV;STD)

##### Paramètres

*Commentaire* : Valeur de probabilité pour laquelle la distribution logarithmique standard doit être calculée, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur centrale dans la distribution normale, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Écart type de la distribution normale, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

NORMINV(0,9;63;5) vaut 69,41

#### 8.1.11.40 NORMSDIST

La fonction NORMDIST() renvoie la distribution normale standard.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

NORMSDIST(Nombre)

##### Paramètres

*Commentaire* : Valeur pour laquelle la distribution normale standard est calculée, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

NORMSDIST(1) vaut 0,84

#### 8.1.11.41 NORMSINV

La fonction NORMSINV() renvoie l'inverse de la distribution cumulative normale standard. Le nombre doit se situer entre 0 et 1 (non inclus).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

NORMSINV(Nombre)

##### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

NORMSINV(0,908789) renvoie 1,333 3

#### 8.1.11.42 PEARSON

La fonction PEARSON() calcule le coefficient de corrélation de deux plages de cellules. Elle est identique à la fonction CORREL.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

PEARSON(plage1 ; plage2)

### Paramètres

*Commentaire* : Plage de valeurs de cellules, *Type* : Double

*Commentaire* : Deuxième plage de valeurs de cellules, *Type* : Double

### Exemples

PEARSON(A1 :A3 ; B1 :B3)

### Fonctions en relation

[CORREL](#)

#### 8.1.11.43 PERCENTILE

La fonction PERCENTILE() renvoie le simple pourcentage x-e de la valeur des données. Un pourcentage renvoie la valeur complète pour une série de données qui vont de la plus petite (alpha=0) à la plus grande (alpha=1) d'une série de données. Pour alpha=25 %, le pourcentage signifie le premier quart; alpha=50 % est le MEDIAN. Les cellules vides sont considérées comme 0, et les cellules avec un texte sont ignorées.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

PERCENTILE(data ;alpha)

### Paramètres

*Commentaire* : Plage de valeurs , *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : La valeur proportionnelle entre 0 et 1 incluse, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Fonctions en relation

[MEDIAN](#)

#### 8.1.11.44 PERMUT

La fonction PERMUT() renvoie le nombre de permutations. Le premier paramètre est le nombre d'éléments et le second est le nombre d'éléments à permuter.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

### Syntaxe

PERMUT(total ;permutés)

### Paramètres

*Commentaire* : Nombre total d'éléments, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombre d'éléments à permuter, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

PERMUT(8;5) vaut 6 720

### Exemples

PERMUT(1;1) vaut 1

#### 8.1.11.45 PERMUTATIONA

La fonction PERMUTATIONA() renvoie le nombre de permutations lorsque les répétitions sont permises. Le premier paramètre est le nombre d'éléments et le second est le nombre d'éléments à permuter. Les deux paramètres doivent être positifs.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

### Syntaxe

PERMUTATIONA(total;choisi)

### Paramètres

*Commentaire* : Nombre total d'éléments, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombre d'éléments à choisir, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

PERMUTATIONA(2,3) renvoie 8

### Exemples

PERMUTATIONA(0,0) renvoie 1

#### 8.1.11.46 PHI

La fonction PHI() renvoie la valeur de la fonction de distribution pour une distribution normale standard.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

PHI(valeur)

### Paramètres

*Commentaire* : Le nombre pour lequel la distribution normale standard doit être calculée,

*Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

PHI(0,25) vaut 0,386 668

#### 8.1.11.47 POISSON

La fonction POISSON() renvoie la distribution de Poisson.

La fonction lambda et les paramètres de nombres doivent être positifs.

Cumulative = 0 calcule la fonction de densité . Cumulative = 1 calcule la distribution.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

POISSON(nombre;lambda;cumul)

### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Paramètre lambda (la valeur du milieu), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : 0 = densité, 1 = distribution, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

POISSON(60;50;0) vaut 0,020 105

### Exemples

POISSON(60;50;1) vaut 0,927 840

#### 8.1.11.48 RANK

La fonction RANK() renvoie le rang d'une nombre dans une liste de nombres.

Ordonner précise comment ranger les nombres :

Si 0 ou omis, les données sont rangées en ordre descendant.

Si pas 0, les données sont rangées en ordre ascendant.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

RSQ(Valeur ; Données ; Ordre)

### Paramètres

*Commentaire* : Valeur, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Données (tableau), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Ordonner, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

RANK (2;{1;2;3}) vaut 2

#### 8.1.11.49 RSQ

La fonction RSQ() renvoie le carré du coefficient de corrélation du moment de production Pearson au travers des points de données dans yConnu et xConnu.

Si "arrayY" et "arrayX" sont vides ou ont un nombre différent de points de données, alors #N/A est renvoyé.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

RSQ( Y connu ; X connu)

### Paramètres

*Commentaire* : Connaître Y (tableau), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Connaître X (tableau), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### 8.1.11.50 SKEW

La fonction SKEW() renvoie une estimation de l'obliquité d'une distribution.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

SKEW(nombre ; nombre2 ; ...)

### Paramètres

*Commentaire* : Flottant, *Type* : Double

*Commentaire* : Flottant, *Type* : Double

*Commentaire* : Flottant, *Type* : Double

*Commentaire* : Flottant, *Type* : Double

### Exemples

SKEW(11,4 ; 17,3 ; 21,3 ; 25,9 ; 40,1) renvoie 0,976 8

### Fonctions en relation

[SKEWP](#)

#### 8.1.11.51 SKEWP

La fonction SKEWP() renvoie l'obliquité de la population d'une distribution

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

SKEWP(nombre ; nombre2 ; ...)

### Paramètres

*Commentaire* : Flottant, *Type* : Double

*Commentaire* : Flottant, *Type* : Double

*Commentaire* : Flottant, *Type* : Double

*Commentaire* : Flottant, *Type* : Double

### Exemples

SKEWP(11,4; 17,3; 21,3; 25,9; 40,1) renvoie 0,655 2

### Fonctions en relation

[SKEW](#)

#### 8.1.11.52 SLOPE

La fonction SLOPE() calcule le slope d'une ligne de régression linéaire.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

SLOPE(y;x)

### Paramètres

*Commentaire* : Valeur y (tableau), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur x (tableau), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### 8.1.11.53 SMALL

La fonction SMALL() renvoie la k-ème plus petite valeur d'un lot de données.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

SMALL(plage; k)

### Paramètres

*Commentaire* : Plage de valeurs de cellules, *Type* : Double

*Commentaire* : Position (depuis le plus petit), *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

A1 : 3, A2 : 1, A3 : 5 => SMALL(A1 :A3; 1) renvoie 1

#### 8.1.11.54 STANDARDIZE

La fonction STANDARDIZE() calcule une valeur normalisée.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

STANDARDIZE(x; moyenne, stdev)

### Paramètres

*Commentaire* : Nombre à normaliser, *Type* : Double

*Commentaire* : Moyenne d'une distribution, *Type* : Double

*Commentaire* : Écart type, *Type* : Double

### Exemples

STANDARDIZE(4; 3; 7) renvoie 0,142 9

#### 8.1.11.55 STDEV

La fonction STDEV() renvoie l'écart-type estimé fondée sur un échantillon. L'écart type est une mesure de la dispersion des valeurs par rapport à la valeur moyenne.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

STDEV(valeur ;valeur ;...)

##### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

STDEV(6;7;8) vaut 1

##### Fonctions en relation

[STDEVP](#)

#### 8.1.11.56 STDEVA

La fonction STDEVA() renvoie l'écart-type estimé sur la base d'un échantillon. L'écart type est une mesure de l'étalement de la dispersion par rapport à la moyenne. Si une cellule référencée contient du texte ou contient la valeur booléenne Faux, elle compte pour 0. Si la valeur booléenne est Vrai, elle est comptée comme 1.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

STDEVA(valeur ;valeur ;...)

##### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

STDEVA(6;7;A1;8) vaut 1, si A1 est vide

##### Exemples

STDEVA(6;7;A1;8) vaut 3,109, si A1 est "Vrai"

##### Fonctions en relation

[STDEV](#)

[STDEVP](#)

### 8.1.11.57 STDEVP

La fonction STDEVP() renvoie l'écart-type fondé sur une population entière.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

STDEVP(valeur ;valeur ;...)

#### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

STDEVP(6;7;8) vaut 0,816 497...

#### Fonctions en relation

[STDEV](#)

### 8.1.11.58 STDEVPA

La fonction STDEVPA() renvoie l'écart-type fondé sur une population entière. Si une cellule référencée contient du texte ou contient la valeur booléenne "Faux", elle compte pour 0. Si la valeur booléenne est "Vrai", elle est comptée comme 1.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

STDEVPA(valeur ;valeur ;...)

#### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

STDEVPA(6 ; 7 ; A1 ; 8) vaut 0,816 497..., si A1 est vide

#### Exemples

STDEVPA(6 ; 7 ; A1 ; 8) vaut 2.69..., si A1 est Vrai

#### Exemples

STDEVPA(6 ; 7 ; A1 ; 8) vaut 3,11..., si A1 est Faux

#### Fonctions en relation

[STDEV](#)

[STDEVP](#)

#### 8.1.11.59 STEYX

La fonction STEYX() calcule l'erreur standard de la valeur y préfixée pour chaque x dans la régression.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SLOPE(y;x)

##### Paramètres

*Commentaire* : Valeur y (tableau), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeur x (tableau), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### 8.1.11.60 SUM2XMY

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SLOPE(y;x)

##### Paramètres

#### 8.1.11.61 SUMPRODUCT

La fonction SUMPRODUCT() (SUM(X\*Y)) renvoie la somme des produits de ces valeurs. Le nombre de valeurs doit être égal entre les deux tableaux. Sinon, la fonction renvoie "Err".

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SUMPRODUCT(tableau1 ;tableau2)

##### Paramètres

*Commentaire* : Valeur (tableau), *Type* : Double

*Commentaire* : Valeur (tableau), *Type* : Double

##### Exemples

SUMPRODUCT(A1 :A2 ;B1 :B2) avec A1=2, A2=5, B1=3 et B2=5 renvoie 31

#### 8.1.11.62 SUMX2MY2

La fonction SUMX2MY2() (SUM(X<sup>2</sup>-Y<sup>2</sup>)) renvoie la différence des carrés de ces valeurs. Le nombre de valeurs doit être égal entre les deux tableaux. Sinon, la fonction renvoie "Err".

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SUMX2MY2(tableau1 ;tableau2)

##### Paramètres

*Commentaire* : Valeur (tableau), *Type* : Double

*Commentaire* : Valeur (tableau), *Type* : Double

##### Exemples

SUMX2MY2(A1 :A2 ;B1 :B2) avec A1=2, A2=5, B1=3 et B2=5 renvoie -5



#### 8.1.11.63 SUMX2PY2

La fonction SUMX2PY2() (SUM(X<sup>2</sup>+Y<sup>2</sup>)) renvoie la somme des carrés de ces valeurs. Le nombre de valeurs doit être égal entre les deux tableaux. Sinon, la fonction renvoie "Err".

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SUMX2PY2(tableau1 ;tableau2)

##### Paramètres

*Commentaire* : Valeur (tableau), *Type* : Double

*Commentaire* : Valeur (tableau), *Type* : Double

##### Exemples

SUMX2PY2(A1 :A2 ;B1 :B2) avec A1=2, A2=5, B1=3 et B2=5 renvoie 63

#### 8.1.11.64 SUMXMY2

La fonction SUMXMY2() (SUM((X-Y)<sup>2</sup>)) renvoie le carré de la différence de ces valeurs. Le nombre de valeurs doit être égal dans les deux tableaux, sinon la fonction renvoie Err .

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SUMXMY2(Tableau1 ;tableau2)

##### Paramètres

*Commentaire* : Valeur (tableau), *Type* : Double

*Commentaire* : Valeur (tableau), *Type* : Double

##### Exemples

SUMXMY2(A1 :A2 ;B1 :B2) avec A1=2, A2=5, B1=3 et B2=5 renvoie 1

#### 8.1.11.65 TDIST

La fonction TDIST() renvoie la distribution t.

Mode = 1 renvoie le test avec une mesure, Mode = 2 renvoie le test avec deux mesures.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

TDIST(nombre ;degrés\_liberté ;mode)

##### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Degrés de liberté pour la distribution t, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Mode (1 ou 2), *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

TDIST(12 ;5 ;1) renvoie 0,000 035

#### 8.1.11.66 TREND

La fonction TREND() calcule une séquence de valeurs fondées sur une régression linéaire de valeurs connues paires.

Contraintes : COUNT(YConnu) = COUNT(XConnu).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

TREND(YConnu[;knownX[;nouveauX[;allowOffset = VRAI]]])

### Paramètres

*Commentaire* : YConnu, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : XConnu, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Nombreséquence du nouveauX, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Offsetpermis, *Type* : Une valeur de vérité (VRAI ou FAUX)

### 8.1.11.67 TRIMMEAN

La fonction TRIMMEAN() calcule la moyenne d'une fraction d'un ensemble de données.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

TRIMMEAN(dataSet; cutOffFraction)

### Paramètres

*Commentaire* : SetDonnées, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : ArrêtFraction, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### 8.1.11.68 TTEST

La fonction TTEST() calcule la probabilité d'un test t.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

TTEST(x; y; type; méthode)

### Paramètres

*Commentaire* : x (tableau), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : y (tableau), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : type, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : méthode, *Type* : Un nombre entier (comme 1, 132, 2344)

### 8.1.11.69 VAR

La fonction VAR() calcule la variance fondée sur un échantillon.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

VAR(valeur;valeur;...)

### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Exemples**

VAR(12;5;7) vaut 13

**Exemples**

VAR(15;80;3) vaut 1 716,333...

**Exemples**

VAR(6;7;8) vaut 1

**Fonctions en relation**

[VARIANCE](#)

[VARA](#)

[VARP](#)

[VARPA](#)

**8.1.11.70 VARA**

La fonction VARA() calcule la variance fondée sur un échantillon.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

VARA(valeur;valeur;...)

**Paramètres**

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Exemples**

VARA(12;5;7) vaut 13

**Exemples**

VARA(15;80;3) vaut 1 716,333...

**Exemples**

VARA(6;7;8) vaut 1

**Fonctions en relation**

[VAR](#)

[VARP](#)

[VARPA](#)

**8.1.11.71 VARIANCE**

La fonction VARIANCE() calcule la variance fondée sur un échantillon. Elle est identique à la fonction VAR.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

VARIANCE(valeur;valeur;...)

### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

VARIANCE(12;5;7) vaut 13

### Exemples

VARIANCE(15;80;3) vaut 1 716,333...

### Exemples

VARIANCE(6;7;8) vaut 1

### Fonctions en relation

[VAR](#)

[VARA](#)

[VARP](#)

[VARPA](#)

### 8.1.11.72 VARP

La fonction VARIANCE() calcule la variance fondée sur une population entière.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Syntaxe

VARP(valeur;valeur;...)

### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

### Exemples

VARP(12;5;7) vaut 8,666...

### Exemples

VARP(15;80;3) vaut 1 144,22...

### Exemples

VARP(6;7;8) vaut 0,666 666 67...

### Fonctions en relation

[VAR](#)

[VARA](#)

[VARPA](#)

### 8.1.11.73 VARPA

La fonction VARPA() calcule la variance fondée sur une population entière. Les valeurs de texte et booléennes qui s'évaluent à Faux sont comptées comme 0, les valeurs booléennes qui s'évaluent à Vrai sont comptées comme 1.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

VARPA(valeur;valeur;...)

#### Paramètres

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Valeurs à virgule flottante, *Type* : Une plage de valeurs à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

VARPA(12;5;7) vaut 8,666...

#### Exemples

VARPA(15;80;3) vaut 1 144,22...

#### Exemples

VARP(6;7;8) vaut 0,666 666 7...

#### Fonctions en relation

[VAR](#)

[VARA](#)

[VARP](#)

### 8.1.11.74 WEIBULL

La fonction WEIBULL() renvoie la distribution de Weibull.

Les paramètres alpha et beta doivent être positifs, le nombre (premier paramètre) ne doit pas être négatif.

Cumulative = 0 calcule la fonction de densité . Cumulative = 1 calcule la distribution.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

WEIBULL(nombre;alpha;beta;cumul)

#### Paramètres

*Commentaire* : Nombre, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Paramètres alpha, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Paramètres beta, *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : 0 = densité, 1 = distribution, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

WEIBULL(2;1;1;0) vaut 0,135 335

#### Exemples

WEIBULL(2;1;1;1) vaut 0,864 665



**Paramètres**

*Commentaire* : Code de caractère, *Type* : Un nombre entier (comme 1, 132, 2344)

**Exemples**

CHAR(65) renvoie "A"

**Fonctions en relation**

[CODE](#)

**8.1.12.4 CLEAN**

La fonction CLEAN() enlève tous les caractères non imprimables d'une chaîne

*Type de retour* : Texte

**Syntaxe**

CLEAN(texte)

**Paramètres**

*Commentaire* : Chaîne source, *Type* : Texte

**Exemples**

CLEAN(AsciiToChar(7) + "HELLO") renvoie "HELLO"

**8.1.12.5 CODE**

La fonction CODE() renvoie un code numérique pour le premier caractère dans une chaîne de texte.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

**Syntaxe**

CODE(texte)

**Paramètres**

*Commentaire* : Texte, *Type* : Texte

**Exemples**

CODE("KDE") renvoie 75

**Fonctions en relation**

[CHAR](#)

**8.1.12.6 COMPARE**

La fonction COMPARE() renvoie 0 si ces deux chaînes sont identiques; -1 si la première est plus basse en valeur que la deuxième; sinon, elle renvoie 1.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

**Syntaxe**

COMPARE(chaîne1;chaîne2; vrai | faux)

**Paramètres**

*Commentaire* : Première chaîne, *Type* : Texte

*Commentaire* : Chaînes à comparer, *Type* : Texte

*Commentaire* : Comparaison sensible à la casse (vrai / faux)?, *Type* : Une valeur de vérité (VRAI ou FAUX)

**Exemples**

COMPARE("Calligra";"Calligra";true) renvoie 0

**Exemples**

COMPARE("calligra";"Calligra";true) renvoie 1

**Exemples**

COMPARE("kspread";"Calligra";false) renvoie 1

**Fonctions en relation**

[EXACT](#)

#### 8.1.12.7 CONCATENATE

La fonction CONCATENATE() renvoie une chaîne qui est la concaténation des chaînes passées en paramètres.

*Type de retour* : Texte

##### Syntaxe

CONCATENATE(valeur;valeur;...)

##### Paramètres

*Commentaire* : Valeurs des chaînes, *Type* : Une plage de chaînes

*Commentaire* : Valeurs des chaînes, *Type* : Une plage de chaînes

*Commentaire* : Valeurs des chaînes, *Type* : Une plage de chaînes

*Commentaire* : Valeurs des chaînes, *Type* : Une plage de chaînes

*Commentaire* : Valeurs des chaînes, *Type* : Une plage de chaînes

##### Exemples

CONCATENATE("Sheets";"Calligra";"KDE") renvoie "SheetsCalligraKDE"

#### 8.1.12.8 DOLLAR

La fonction DOLLAR() convertit un nombre en texte en utilisant le format monétaire, avec les décimales arrondies à l'endroit spécifié. Bien que le nom soit DOLLAR, cette fonction fera la conversion avec les unités locales.

*Type de retour* : Texte

##### Syntaxe

DOLLAR(nombre;décimales)

##### Paramètres

*Commentaire* : Nombre, *Type* : Double

*Commentaire* : Décimales, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

DOLLAR(1403,77) renvoie "1 403,77 €"

##### Exemples

DOLLAR(-0,123;4) renvoie "0,123 0 €"

#### 8.1.12.9 EXACT

La fonction EXACT() renvoie "Vrai" si ces deux chaînes sont identiques, sinon elle renvoie "Faux".

*Type de retour* : Une valeur de vérité (VRAI ou FAUX)

##### Syntaxe

EXACT(chaîne1;chaîne2)

##### Paramètres

*Commentaire* : Chaîne, *Type* : Texte

*Commentaire* : Chaîne, *Type* : Texte

##### Exemples

EXACT("Calligra";"Calligra") renvoie Vrai

##### Exemples

EXACT("KSpread";"Calligra") returns Faux

##### Fonctions en relation

[COMPARE](#)



#### 8.1.12.10 FIND

La fonction FIND() cherche une chaîne de texte (texte\_cherché) dans une autre chaîne de texte (dans\_le\_texte) et renvoie le numéro du point de départ du "texte\_cherché", depuis le caractère le plus à gauche de "dans\_le\_texte".

Le paramètre "num\_départ" spécifie le caractère où la recherche commencera. Le premier caractère est le caractère numéro 1. Si "num\_départ" est omis, on suppose qu'il vaut 1.

Vous pouvez aussi utiliser la fonction "SEARCH", mais contrairement à elle, "FIND" tient compte de la casse et n'autorise pas de jokers.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

FIND(texte\_cherché;dans\_le\_texte;num\_départ)

##### Paramètres

*Commentaire* : Le texte que vous voulez trouver, *Type* : Texte

*Commentaire* : Le texte qui peut contenir "texte\_cherché", *Type* : Texte

*Commentaire* : Spécifie l'index où commencer la recherche, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

FIND("Cal";"Calligra") renvoie 1

##### Exemples

FIND("i";"Calligra") renvoie 5

##### Exemples

FIND("a";"Sheets in Calligra";4) renvoie 12

##### Fonctions en relation

[FINDB](#)  
[SEARCH](#)  
[REPLACE](#)  
[SEARCHB](#)  
[REPLACEB](#)

#### 8.1.12.11 FINDB

La fonction FINDB() cherche une chaîne de texte (texte\_cherché) dans une autre chaîne de texte (dans\_le\_texte) et renvoie le numéro du point de départ du « texte\_cherché », depuis le caractère le plus à gauche de « dans\_le\_texte » en utilisant la position des octets.

Le paramètre PositionOctet spécifie le caractère où la recherche commencera. Le premier caractère est le caractère numéro 2. Si « num\_départ » est omis, on suppose qu'il vaut 2.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

FINDB(texte\_cherché;dans\_le\_texte;départ PositionOctet)

##### Paramètres

*Commentaire* : Le texte que vous voulez trouver, *Type* : Texte

*Commentaire* : Le texte qui peut contenir "texte\_cherché", *Type* : Texte

*Commentaire* : Spécifie la position de l'octet où commencer la recherche, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Fonctions en relation

[FIND](#)  
[SEARCH](#)  
[REPLACE](#)  
[SEARCHB](#)  
[REPLACEB](#)

#### 8.1.12.12 FIXED

La fonction FIXED() arrondit un nombre au nombre spécifié de décimales, formate le nombre en chaîne de format décimal et renvoie le résultat en texte. Si "Décimales" est négatif, le nombre est arrondi à gauche du séparateur décimal. Si vous omettez "Décimales", il est supposé valoir 2. Si le paramètre optionnel "Pas\_de\_séparateur" est "Vrai", le séparateur des milliers ne sera pas affiché.

*Type de retour* : Texte

##### Syntaxe

FIXED(nombre;décimales;pas\_de\_séparateur)

##### Paramètres

*Commentaire* : Nombre, *Type* : Double

*Commentaire* : Décimales, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Pas\_de\_séparateur, *Type* : Une valeur de vérité (VRAI ou FAUX)

##### Exemples

FIXED(1234,567;1) renvoie "1 234,6"

##### Exemples

FIXED(1234,567;1;FAUX) renvoie "1234,6"

##### Exemples

FIXED(44,332) renvoie "44,33"

#### 8.1.12.13 JIS

La fonction JIS() renvoie les caractères extra-larges correspondant à l'argument demi-large.

*Type de retour* : Texte

##### Syntaxe

JIS(texte)

##### Paramètres

*Commentaire* : Caractères demi-large, *Type* : Texte

##### Fonctions en relation

[ASC](#)

#### 8.1.12.14 LEFT

La fonction LEFT() renvoie une sous-chaîne qui contient la longueur des caractères les plus à gauche de la chaîne. La chaîne complète est renvoyée si cette longueur dépasse la longueur de la chaîne. Une erreur est indiquée si le nombre de caractères est inférieur à 0.

*Type de retour* : Texte

##### Syntaxe

LEFT(texte;longueur)

##### Paramètres

*Commentaire* : Chaîne source, *Type* : Texte

*Commentaire* : Nombre de caractères, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

LEFT("HELLO";2) renvoie "he"

##### Exemples

LEFT("KSpread";10) renvoie "KSpread"

### Exemples

LEFT("KSpread") renvoie "K"

### Fonctions en relation

RIGHT  
MID  
RIGHTB  
MIDB

#### 8.1.12.15 LEFTB

La fonction LEFTB() renvoie une sous-chaîne qui contient la longueur des caractères les plus à gauche de la chaîne en utilisant la position des octets. La chaîne complète est renvoyée si cette longueur dépasse la longueur de la chaîne. Une erreur est indiquée si le nombre de caractères est inférieur à 0.

*Type de retour* : Texte

### Syntaxe

LEFTB(texte;longueur en octet)

### Paramètres

*Commentaire* : Chaîne source, *Type* : Texte

*Commentaire* : Longueur en octet, *Type* : Un nombre entier (comme 1, 132, 2344)

### Fonctions en relation

RIGHT  
MID  
RIGHTB  
MIDB

#### 8.1.12.16 LEN

La fonction LEN() renvoie la longueur d'une chaîne.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

### Syntaxe

LEN(texte)

### Paramètres

*Commentaire* : Chaîne, *Type* : Texte

### Exemples

LEN("HELLO") renvoie 5

### Exemples

LEN("KSpread") renvoie 7

### Fonctions en relation

LENB

#### 8.1.12.17 LENB

La fonction LENB() renvoie la longueur de la chaîne en utilisant la position des octets.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

### Syntaxe

LENB(texte)

### Paramètres

*Commentaire* : Chaîne, *Type* : Texte

#### 8.1.12.18 LOWER

La fonction LOWER() convertit une chaîne en minuscules.

*Type de retour* : Texte

##### Syntaxe

LOWER(texte)

##### Paramètres

*Commentaire* : Chaîne source, *Type* : Texte

##### Exemples

LOWER("HELLO") renvoie "hello"

##### Exemples

LOWER("HELLO") renvoie "hello"

##### Fonctions en relation

[UPPER](#)

[TOGGLE](#)

#### 8.1.12.19 MID

La fonction MID() renvoie une sous-chaîne qui contient les "longueur" caractères de la chaîne, débutant à l'index "position".

*Type de retour* : Texte

##### Syntaxe

MID(texte;position;longueur)

##### Paramètres

*Commentaire* : Chaîne source, *Type* : Texte

*Commentaire* : Position, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Longueur, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

MID("Calligra";2;3) renvoie "all"

##### Exemples

MID("Calligra";2) renvoie "alligra"

##### Fonctions en relation

[LEFT](#)

[RIGHT](#)

[LEFTB](#)

[RIGHTB](#)

[MIDB](#)

#### 8.1.12.20 MIDB

La fonction MIDB() renvoie une sous-chaîne qui contient les « longueur » caractères de la chaîne, débutant à l'index « position », en utilisant la position des octets.

*Type de retour* : Texte

##### Syntaxe

MIDB(texte;position en octets;longueur en octets)

##### Paramètres

*Commentaire* : Chaîne source, *Type* : Texte

*Commentaire* : Position en octet, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Longueur en octet, *Type* : Un nombre entier (comme 1, 132, 2344)

### Fonctions en relation

LEFT  
RIGHT  
LEFTB  
RIGHTB  
MID

#### 8.1.12.21 PROPER

La fonction PROPER convertit la première lettre de chaque mot en majuscule et le reste en minuscule.

*Type de retour* : Texte

##### Syntaxe

PROPER(chaîne)

##### Paramètres

*Commentaire* : Chaîne, *Type* : Texte

##### Exemples

PROPER("ceci est un titre") renvoie "Ceci Est Un Titre"

#### 8.1.12.22 REGEXP

Renvoie la partie d'une chaîne correspondant à une expression rationnelle. Si la chaîne ne correspond pas à l'expression rationnelle donnée, la valeur spécifiée dans défaut est renvoyée.

Si une référence arrière est fournie, alors la valeur de cette référence arrière est renvoyée.

Si le paramètre défaut n'est pas précisé, on suppose que c'est la chaîne vide. Si aucune référence arrière n'est donnée, on suppose que c'est 0 (toute la partie correspondante est ainsi renvoyée).

*Type de retour* : Texte

##### Syntaxe

REGEXP(texte; exp\_rationnelle; défaut; réf\_arrière)

##### Paramètres

*Commentaire* : Texte recherché, *Type* : Texte

*Commentaire* : Expression rationnelle, *Type* : Texte

*Commentaire* : Valeur par défaut (optionnel), *Type* : Texte

*Commentaire* : Référence arrière (optionnel), *Type* : Nombre

##### Exemples

REGEXP("Le nombre est 15.";"[0-9]+") = "15"

##### Exemples

REGEXP("15, 20, 26, 41";"([0-9]+), \*[0-9]+\$" ";" ;1) = "26"

#### 8.1.12.23 REGEXPRE

Remplace toutes les parties correspondant à l'expression rationnelle avec le texte de remplacement

*Type de retour* : Texte

##### Syntaxe

REGEXPRE(texte; exp\_rationnelle; remplacement)

##### Paramètres

*Commentaire* : Texte recherché, *Type* : Texte

*Commentaire* : Expression rationnelle, *Type* : Texte

*Commentaire* : Remplacement, *Type* : Texte

##### Exemples

REGEXPRE("14 et 15 et 16";"[0-9]+"; "num") renvoie "num et num et num"

#### 8.1.12.24 REPLACE

La fonction REPLACE() remplace une partie d'une chaîne de texte par une chaîne différente.

*Type de retour* : Texte

##### Syntaxe

REPLACE(texte;position;longueur;nouveau\_texte)

##### Paramètres

*Commentaire* : Texte que vous voulez remplacer par des caractères, *Type* : Texte

*Commentaire* : Position des caractères à remplacer, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Nombre de caractères à remplacer, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Le texte qui remplacera les caractères dans l'ancien texte, *Type* : Texte

##### Exemples

REPLACE("abcdefghijkl" ;6;5;"-") renvoie "abcde-k"

##### Exemples

REPLACE("2002" ;3;2;"03 x) renvoie "2003"

##### Fonctions en relation

[FIND](#)

[MID](#)

[FINDB](#)

[MIDB](#)

#### 8.1.12.25 REPLACEB

La fonction REPLACEB() remplace une partie d'une chaîne de texte par une chaîne différente en utilisant la position des octets.

*Type de retour* : Texte

##### Syntaxe

REPLACEB(texte;position en octets;longueur en octets,nouveau texte)

##### Paramètres

*Commentaire* : Le texte que vous voulez remplacer par des caractères en utilisant la position des octets, *Type* : Texte

*Commentaire* : Position des octets des caractères à remplacer, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Longueur en octets des caractères à remplacer, *Type* : Un nombre entier (comme 1, 132, 2344)

*Commentaire* : Le texte qui remplacera les caractères dans l'ancien texte, *Type* : Texte

##### Fonctions en relation

[FINDB](#)

[MIDB](#)

[FIND](#)

[MID](#)

#### 8.1.12.26 REPT

La fonction REPT() répète le premier paramètre autant de fois qu'indiqué par le second paramètre. Le second paramètre ne doit pas être négatif, et cette fonction retourne une chaîne vide si ce deuxième paramètre est à zéro (ou arrondi à zéro).

*Type de retour* : Texte

### Syntaxe

REPT(texte;compte)

### Paramètres

*Commentaire* : Chaîne source, *Type* : Texte

*Commentaire* : Nombre de répétitions, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

REPT("KSpread";3) renvoie "KSpreadKSpreadKSpread"

### Exemples

REPT("KSpread";0) renvoie ""

### 8.1.12.27 RIGHT

La fonction RIGHT() renvoie une sous-chaîne qui contient les "longueur" caractères les plus à droite de la chaîne. La chaîne complète est retournée si "longueur" dépasse la longueur de la chaîne.

*Type de retour* : Texte

### Syntaxe

RIGHT(texte;longueur)

### Paramètres

*Commentaire* : Chaîne source, *Type* : Texte

*Commentaire* : Nombre de caractères, *Type* : Un nombre entier (comme 1, 132, 2344)

### Exemples

RIGHT("HELLO";2) renvoie "lo"

### Exemples

LEFT("KSpread";10) renvoie "KSpread"

### Exemples

RIGHT("KSpread") renvoie "d"

### Fonctions en relation

[LEFT](#)

[MID](#)

[LEFTB](#)

[MIDB](#)

### 8.1.12.28 RIGHTB

La fonction RIGHTB() renvoie une sous-chaîne qui contient les « longueur » caractères les plus à droite de la chaîne en utilisant les positions des octets. La chaîne complète est retournée si « longueur » dépasse la longueur de la chaîne.

*Type de retour* : Texte

### Syntaxe

RIGHTB(texte;longueur en octet)

### Paramètres

*Commentaire* : Chaîne source, *Type* : Texte

*Commentaire* : Longueur en octet, *Type* : Un nombre entier (comme 1, 132, 2344)

### Fonctions en relation

[LEFT](#)

[MID](#)

[LEFTB](#)

[MIDB](#)

#### 8.1.12.29 ROT13

La fonction ROT13() chiffre le texte en remplaçant chaque lettre par celle qui se trouve 13 places plus loin dans l'alphabet. Si la 13ème position est au-delà de la lettre Z, on recommence à A (rotation).

En appliquant la fonction de chiffrement à nouveau, le texte résultant est le texte déchiffré.

*Type de retour* : Texte

##### Syntaxe

ROT13(Texte)

##### Paramètres

*Commentaire* : Texte, *Type* : Texte

##### Exemples

ROT13("KSpread") renvoie "XFcernq"

##### Exemples

ROT13("XFcernq") renvoie "KSpread"

#### 8.1.12.30 SEARCH

La fonction SEARCH() cherche une chaîne de texte (texte\_cherché) dans une autre chaîne de texte (dans\_le\_texte) et renvoie le nombre du point de démarrage de "texte\_cherché", depuis le caractère le plus à gauche de "dans\_le\_texte".

Vous pouvez utiliser les jokers, points d'interrogation (?) et astérisque (\*). Un point d'interrogation correspond à un seul caractère, un astérisque à n'importe quelle séquence de caractères.

Le paramètre "num\_début" spécifie le caractère où commencer la recherche. Le premier caractère est le numéro 1. Si "num\_début" manque, il est supposé valoir 1. "SEARCH" ne distingue pas les majuscules des minuscules.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

##### Syntaxe

SEARCH(texte\_cherché;dans\_le\_texte;num\_début)

##### Paramètres

*Commentaire* : Le texte que vous voulez trouver, *Type* : Texte

*Commentaire* : Le texte qui peut contenir "texte\_cherché", *Type* : Texte

*Commentaire* : Index spécifié pour le début de la recherche, *Type* : Un nombre entier (comme 1, 132, 2344)

##### Exemples

SEARCH("e";"Statements";6) renvoie 7

##### Exemples

SEARCH("margin";"Profit Margin") renvoie 8

##### Fonctions en relation

[FIND](#)

[FINDB](#)

[SEARCHB](#)

#### 8.1.12.31 SEARCHB

La fonction SEARCHB() cherche une chaîne de texte (texte\_cherché) dans une autre chaîne de texte (dans\_le\_texte) et renvoie le nombre du point de démarrage de « texte\_cherché », depuis le caractère le plus à gauche de « dans\_le\_texte » en utilisant les positions des octets.



Vous pouvez utiliser les jokers, points d'interrogation (?) et astérisque (\*). Un point d'interrogation correspond à un seul caractère, un astérisque à n'importe quelle séquence de caractères.

Le paramètre PositionOctet spécifie le caractère où commencer la recherche. Le premier caractère est le numéro 2. Si PositionOctet manque, il est supposé valoir 2. SEARCHB ne distingue pas les majuscules des minuscules.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

#### Syntaxe

SEARCHB(texte\_cherché;dans\_le\_texte;départ PositionOctet)

#### Paramètres

*Commentaire* : Le texte que vous voulez trouver, *Type* : Texte

*Commentaire* : Le texte qui peut contenir "texte\_cherché", *Type* : Texte

*Commentaire* : Position de l'octet donné pour le début de la recherche, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Fonctions en relation

FINDB  
FIND  
SEARCH

#### 8.1.12.32 SLEEK

La fonction SLEEK() enlève toutes les espaces d'une chaîne.

*Type de retour* : Texte

#### Syntaxe

SLEEK(texte)

#### Paramètres

*Commentaire* : Chaîne source, *Type* : Texte

#### Exemples

SLEEK("Voici du texte") renvoie "Voicidutexte"

#### Fonctions en relation

TRIM

#### 8.1.12.33 SUBSTITUTE

La fonction SUBSTITUTE() substitue « nouveau\_texte » à « ancien\_texte » dans une chaîne de texte. Si « num\_d\_instance » est spécifié, seule cette instance de l'ancien texte sera remplacée. Sinon, chaque occurrence d'« ancien\_texte » sera changée en « nouveau\_texte ». Utilisez « SUBSTITUTE » quand vous voulez remplacer du texte spécifique, utilisez « REPLACE » quand vous voulez remplacer du texte qui se trouve à un endroit précis.

*Type de retour* : Texte

#### Syntaxe

SUBSTITUTE(texte;ancien\_texte;nouveau\_texte;num\_d\_instance)

#### Paramètres

*Commentaire* : Texte pour lequel vous voulez substituer, *Type* : Texte

*Commentaire* : Partie du texte que vous voulez remplacer, *Type* : Texte

*Commentaire* : Nouveau texte qui sera le remplacement, *Type* : Texte

*Commentaire* : Occurrence à remplacer, *Type* : Un nombre entier (comme 1, 132, 2344)

#### Exemples

SUBSTITUTE("Cost Data";"Cost";"Sales") renvoie "Sales Data"

**Exemples**

SUBSTITUTE("Qtr 1, 2001";"1";"3";1) renvoie "Qtr 3, 2001"

**Exemples**

SUBSTITUTE("Qtr 1, 2001";"1";"3";4) renvoie "Qtr 3, 2003"

**Fonctions en relation**

REPLACE

REPLACEB

FIND

FINDB

**8.1.12.34 T**

La fonction T() renvoie le texte auquel on se réfère par "Valeur". Si "Valeur" est ou se rapporte à du texte, T renvoie "Valeur". Si "Valeur" ne se rapporte pas à du Texte, T renvoie du texte vide.

*Type de retour* : Texte

**Syntaxe**

T(valeur)

**Paramètres**

*Commentaire* : Valeur, *Type* : N'importe quel type de valeur

**Exemples**

T("Calligra") renvoie "Calligra"

**Exemples**

T(1,2) renvoie "" (texte vide)

**8.1.12.35 TEXT**

La fonction TEXT() convertit une valeur en texte.

*Type de retour* : Texte

**Syntaxe**

TEXT(valeur)

**Paramètres**

*Commentaire* : Valeur, *Type* : N'importe quel type de valeur

**Exemples**

TEXT(1234,56) renvoie "1234,56"

**Exemples**

TEXT("KSpread";10) renvoie "KSpread"

**8.1.12.36 TOGGLE**

La fonction TOGGLE() change les minuscules en majuscules et les majuscules en minuscules.

*Type de retour* : Texte

**Syntaxe**

TOGGLE(texte)

**Paramètres**

*Commentaire* : Chaîne source, *Type* : Texte

**Exemples**

TOGGLE("HELLO") renvoie "HELLO"

**Exemples**

TOGGLE("HELLO") renvoie "hello"

**Exemples**

TOGGLE("HELLO") renvoie "hELLo"

**Fonctions en relation**

[UPPER](#)

[LOWER](#)

**8.1.12.37 TRIM**

La fonction TRIM() renvoie du texte avec un seul espace entre les mots.

*Type de retour* : Texte

**Syntaxe**

TRIM(texte)

**Paramètres**

*Commentaire* : Chaîne, *Type* : Texte

**Exemples**

TRIM(" hello KSpread ") renvoie "hello KSpread"

**8.1.12.38 UNICHAR**

La fonction UNICHAR() renvoie le caractère spécifié par un point de code numérique.

*Type de retour* : Texte

**Syntaxe**

UNICHAR(code)

**Paramètres**

*Commentaire* : Code de caractère, *Type* : Un nombre entier (comme 1, 132, 2344)

**Exemples**

UNICHAR(65) renvoie "A"

**Fonctions en relation**

[UNICODE](#)

[CHAR](#)

**8.1.12.39 UNICODE**

La fonction UNICODE() renvoie un code numérique pour le premier caractère dans une chaîne de texte.

*Type de retour* : Un nombre entier (comme 1, 132, 2344)

**Syntaxe**

UNICODE(texte)

**Paramètres**

*Commentaire* : Texte, *Type* : Texte

**Exemples**

UNICODE("KDE") renvoie 75

**Fonctions en relation**

[UNICHAR](#)

[CODE](#)

#### 8.1.12.40 UPPER

La fonction UPPER() convertit une chaîne en majuscules.

*Type de retour* : Texte

##### Syntaxe

UPPER(texte)

##### Paramètres

*Commentaire* : Chaîne source, *Type* : Texte

##### Exemples

UPPER("HELLO") renvoie "HELLO"

##### Exemples

UPPER("HELLO") renvoie "HELLO"

##### Fonctions en relation

LOWER  
TOGGLE

#### 8.1.12.41 VALUE

Convertit une chaîne de texte qui représente une valeur en la valeur réelle.

*Type de retour* : Double

##### Syntaxe

VALUE(texte)

##### Paramètres

*Commentaire* : Texte, *Type* : Texte

##### Exemples

VALUE("14,03") renvoie 14,03

### 8.1.13 Trigonométrie

#### 8.1.13.1 ACOS

La fonction ACOS() renvoie l'arc cosinus en radians et la valeur est mathématiquement définie pour se situer entre 0 et PI (inclus).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

ACOS(Flottant)

##### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

ACOS(0,8) vaut 0,643 501 1

##### Exemples

ACOS(0) vaut 1,570 796 33

##### Fonctions en relation

COS

### 8.1.13.2 ACOSH

La fonction ACOSH() calcule le cosinus hyperbolique inverse de x, qui est la valeur dont le cosinus hyperbolique est x. Si x est plus petit que 1, acosh() renvoie NaN (not-a-number) et une erreur.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

ACOSH(Flottant)

#### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

ACOSH(5) vaut 2,292 431 67

#### Exemples

ACOSH(0) vaut "NaN"

#### Fonctions en relation

[COSH](#)

### 8.1.13.3 ACOT

La fonction ACOT() renvoie l'arc cotangente d'un nombre.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

ACOT(Flottant)

#### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

ACOT(0) vaut 1,570 796 33

### 8.1.13.4 ASIN

La fonction ASIN() renvoie l'arc sinus en radians et la valeur est mathématiquement définie pour se situer entre -PI/2 et PI/2 (inclus).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

ASIN(Flottant)

#### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

ASIN(0,8) vaut 0,927 295 22

#### Exemples

ASIN(0) vaut 0

#### Fonctions en relation

[SIN](#)

#### 8.1.13.5 ASINH

La fonction ASINH() calcule le sinus hyperbolique inverse de x, qui est la valeur dont le sinus hyperbolique est x.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

ASINH(Flottant)

##### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

ASINH(0,8) vaut 0,732 668 26

##### Exemples

ASINH(0) vaut 0

##### Fonctions en relation

[SINH](#)

#### 8.1.13.6 ATAN

La fonction ATAN() renvoie l'arc tangente en radians et la valeur est mathématiquement définie pour se situer entre  $-\pi/2$  et  $\pi/2$  (inclus).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

ATAN(Flottant)

##### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

ATAN(0,8) vaut 0,674 740 94

##### Exemples

ATAN(0) vaut 0

##### Fonctions en relation

[TAN](#)

[ATAN2](#)

#### 8.1.13.7 ATAN2

Cette fonction calcule l'arc-tangente de deux variables x et y. Elle est semblable au calcul de l'arc-tangente de  $y / x$ , sauf que les signes des deux arguments sont utilisés pour déterminer le quadrant du résultat.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

ATAN2(valeur;valeur)

##### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Exemples**

ATAN2(0,5;1) vaut 1,107 149

**Exemples**

ATAN2(-0,5;2,0) vaut 1,815 775

**Fonctions en relation**

[ATAN](#)

**8.1.13.8 ATANH**

La fonction ATANH() calcule la tangente hyperbolique inverse de x, qui est la valeur dont la tangente hyperbolique est x. Si la valeur absolue de x est plus grande que 1, ATANH() renvoie NaN (not-a-number).

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

ATANH(Flottant)

**Paramètres**

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Exemples**

ATANH(0,8) vaut 1,098 612 29

**Exemples**

ATANH(0) vaut 0

**Fonctions en relation**

[TANH](#)

**8.1.13.9 COS**

La fonction COS() renvoie le cosinus de x, avec x en radians.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Syntaxe**

COS(Flottant)

**Paramètres**

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

**Exemples**

COS(0) vaut 1

**Exemples**

COS(PI()/2) vaut 0

**Fonctions en relation**

[SIN](#)

[ACOS](#)

#### 8.1.13.10 COSH

La fonction COSH() renvoie le cosinus hyperbolique de  $x$ , qui est défini mathématiquement comme  $(\exp(x) + \exp(-x)) / 2$ .

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

COSH(Flottant)

##### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

COSH(0,8) vaut 1,337 434 95

##### Exemples

COSH(0) vaut 1

##### Fonctions en relation

[ACOSH](#)

#### 8.1.13.11 CSC

La fonction CSC() renvoie la cosécante de  $x$ , avec  $x$  en radians.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

CSC(réel)

##### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

CSC(PI()/2) vaut 1

#### 8.1.13.12 CSCH

La fonction CSCH() renvoie la cosécante hyperbolique de  $x$ , avec  $x$  en radians.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

CSCH(réel)

##### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

CSCH(PI()/2) vaut 0.434537208...



### 8.1.13.13 DEGREES

Cette fonction transforme un angle en radians en angle en degrés.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

DEGREES(Flottant)

#### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

DEGREES(0,78) vaut 44,69

#### Exemples

DEGREES(1) vaut 57,29

#### Fonctions en relation

[RADIANS](#)

### 8.1.13.14 PI

La fonction PI() renvoie la valeur de PI.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

PI()

#### Paramètres

#### Exemples

PI() vaut 3,141 592 65...

### 8.1.13.15 RADIANS

Cette fonction transforme un angle en degrés en un angle en radians.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Syntaxe

RADIANS(Flottant)

#### Paramètres

*Commentaire* : Angle (degrés), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

#### Exemples

RADIANS(75) vaut 1,308

#### Exemples

RADIANS(90) vaut 1,570 7

#### Fonctions en relation

[DEGREES](#)

#### 8.1.13.16 SEC

La fonction SEC() renvoie la sécante de  $x$ , avec  $x$  en radians.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SEC(Float)

##### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

SEC(0) vaut 1

#### 8.1.13.17 SECH

La fonction SECH() renvoie la sécante hyperbolique de  $x$ , avec  $x$  en radians.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SECH(Float)

##### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

SECH(0) vaut 1

#### 8.1.13.18 SIN

La fonction SIN() renvoie le sinus de  $x$ , avec  $x$  en radians.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SIN(Flottant)

##### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

SIN(0) vaut 0

##### Exemples

SIN(PI()/2) vaut 1

##### Fonctions en relation

[COS](#)

[ASIN](#)

#### 8.1.13.19 SINH

La fonction SINH() renvoie le sinus hyperbolique de  $x$ , qui est défini mathématiquement comme  $(\exp(x) - \exp(-x)) / 2$ .

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

SINH(Flottant)

##### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

SINH(0,8) vaut 0,888 105 98

##### Exemples

SINH(0) vaut 0

##### Fonctions en relation

[ASINH](#)

#### 8.1.13.20 TAN

La fonction TAN() renvoie la tangente de  $x$ , avec  $x$  en radians.

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

TAN(Flottant)

##### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

TAN(0,7) vaut 0,842 288 38

##### Exemples

TAN(0) vaut 0

##### Fonctions en relation

[ATAN](#)

#### 8.1.13.21 TANH

La fonction TANH() renvoie la tangente hyperbolique de  $x$ , qui est définie mathématiquement comme  $\sinh(x) / \cosh(x)$ .

*Type de retour* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Syntaxe

TANH(Flottant)

##### Paramètres

*Commentaire* : Angle (radians), *Type* : Une valeur à virgule flottante (comme 1,3 ou 0,343 ou 253)

##### Exemples

TANH(0,8) vaut 0,664 036 77

##### Exemples

TANH(0) vaut 0

##### Fonctions en relation

[ATANH](#)

## Chapitre 9

# Remerciements et licence

Calligra Sheets

Droits d’auteur pour le programme 1998 - 2019 par l’équipe de Calligra Sheets :

- Torben Weis [weis@kde.org](mailto:weis@kde.org)
- Laurent Montel [lmontel@mandrakesoft.com](mailto:lmontel@mandrakesoft.com)
- David Faure [faure@kde.org](mailto:faure@kde.org)
- John Dailey [dailey@vt.edu](mailto:dailey@vt.edu)
- Philipp Müller [philipp.mueller@gmx.de](mailto:philipp.mueller@gmx.de)
- Ariya Hidayat [ariya@kde.org](mailto:ariya@kde.org)
- Norbert Andres [nandres@web.de](mailto:nandres@web.de)
- Shaheed Haque [srhaque@iee.org](mailto:srhaque@iee.org)
- Werner Trobin [trobin@kde.org](mailto:trobin@kde.org)
- Nikolas Zimmermann [wildfox@kde.org](mailto:wildfox@kde.org)
- Helge Deller [deller@kde.org](mailto:deller@kde.org)
- Percy Leonhart [percy@eris23.org](mailto:percy@eris23.org)
- Eva Brucherseifer [eva@kde.org](mailto:eva@kde.org)
- Phillip Ezolt [phillipezolt@hotmail.com](mailto:phillipezolt@hotmail.com)
- Enno Bartels [ebartels@nwn.de](mailto:ebartels@nwn.de)
- Graham Short [grahshrt@netscape.net](mailto:grahshrt@netscape.net)

Documentation copyright 2002 Pamela Roberts [pamroberts@blueyonder.co.uk](mailto:pamroberts@blueyonder.co.uk)

Mises à jour mineures de la documentation pour KOffice 1.3 par Philip Rodrigues [phil@kde.org](mailto:phil@kde.org).

Mises à jour des copies d’écran pour Calligra 3.1 par Carl Schwan [carl@carlschwan.eu](mailto:carl@carlschwan.eu)

Traduction française par GerardDelafond [gerard@delafond.org](mailto:gerard@delafond.org), Daniel Durand [ddurand42@gmail.com](mailto:ddurand42@gmail.com) et Ludovic Grossard [grossard@kde.org](mailto:grossard@kde.org).

Cette documentation est soumise aux termes de la Licence de Documentation Libre GNU (GNU Free Documentation License).

Ce programme est soumis aux termes de la Licence Générale Publique GNU (GNU General Public License).