

Copyright Information

Copyright 2001, Investment Intelligence Systems Corporation. All Rights Reserved.

The information contained in this manual and accompanying software program is copyrighted and all its rights are reserved by Investment Intelligence Systems Corporation (IISC). IISC reserves the right to make periodic modifications of this product without obligation to notify any person or entity of such revision. Copying, duplicating, selling, or otherwise distributing any part of this product without the prior consent of an authorized representative of IISC is prohibited.

JSheet and HyperSheet are registered trademarks of Investment Intelligence Systems Corporation.

Disclaimer of Warranties

The software and users manuals are provided “as is” and without express or limited warranty of any kind by either IISC or anyone who has been involved in the creation, production, or distribution of the software, including, but not limited to the implied warranties of the merchantability and fitness for a particular purpose. The entire risk as to quality and performance of the software and users manuals is with you. Should the software and users manuals prove defective, you (and IISC or anyone else who has been involved in the creation, production, or distribution of the software) assume the entire cost of all necessary servicing, repairs, or correction.

Some states do not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

Limitation of Liability

In no event will IISC or any other person involved in the creation, production, or distribution of the software be liable to you on account of any claim for any damages, including any lost profits, lost savings, or other special, incidental, consequential, or exemplary damages, including but not limited to any damages assessed against or paid by you to any third party, arising out of the use, inability to use, quality, or performance of such software and users manuals, even if IISC or any other such person or entity has been advised of the possibility for such damages, or for any claim by any party. In addition, IISC or any other person involved in the creation, production, or distribution of the software shall not be liable for any claim by you or any other party for damages arising out

of the use, inability to use, quality, or performance of such software and users manuals, based upon principals of contract warranty, negligence, strict liability for the negligence of IISC or other tort, breach of any statutory duty, principals of indemnity or contribution, the failure of any remedy to achieve its essentials purpose, or otherwise.

Some states do not allow the limitation or exclusion of liability for incidental or consequential damages, so the above limitation may not apply to you.

Table of Contents

CHAPTER 1: INTRODUCING FUNCTIONS	1-1
Function Keywords	1-1
Using Number Arguments.....	1-2
Using Dates and Times as Arguments.....	1-2
Using String Arguments.....	1-3
Using Range Arguments	1-3
Types of Worksheet Ranges.....	1-4
Using Linked Ranges as Arguments	1-4
Using Optional Function Arguments.....	1-5
Using Formulas as Arguments	1-5
Using Operators in a Formula	1-5
The Evaluation Order in Formulas	1-7
CHAPTER 2: FUNCTIONS BY GROUP	2-1
Numeric Functions	2-1
Financial Functions	2-2
Statistical Functions	2-4
Engineering Functions	2-5
Logical Functions	2-5
Database Functions	2-6
Text Functions	2-7

Date and Time Functions.....	2-7
Color Functions	2-8
Worksheet Functions	2-8
Worksheet Status Functions.....	2-9
CHAPTER 3: ALPHABETICAL LIST OF FUNCTIONS	3-1
ABS	3-1
ACCRINT	3-2
ACCRINTM.....	3-2
ACOS	3-3
ACOSH	3-4
ADATE	3-4
ADDDAYS	3-5
ADDHOURS.....	3-6
ADDMINUTES.....	3-7
ADDMONTHS	3-7
ADDSECONDS	3-8
ADDYEARS	3-9
AMORDEGRC	3-9
AMORLINC.....	3-10
ASIN.....	3-11
ASINH.....	3-11
ATAN.....	3-12
ATAN2.....	3-12
ATANH	3-13
ATIME	3-13
AVEDEV.....	3-14
AVG	3-15
BESSELI	3-16
BESSELJ	3-16
BESSELK.....	3-17
BESSELY.....	3-17
BETADIST.....	3-18

BETAINV	3-18
BIN2DEC	3-19
BIN2HEX.....	3-19
BIN2OCT	3-20
BINOMDIST	3-20
BITAND.....	3-21
BITOR.....	3-22
BITXOR.....	3-24
BLACK	3-25
BLUE.....	3-25
BNDCMPXPRICE	3-26
BNDCMPXYIELD	3-26
BNDCONVEXITY	3-27
BNDDURATION.....	3-27
BNDESTPRICE	3-28
BNDPRICE	3-28
BNDPRICECHANGE.....	3-29
BNDSENSITIVITY	3-29
BNDYIELD.....	3-30
BONDPRICE	3-30
BONDYTM.....	3-31
BOOKOF.....	3-31
BRIGHTNESS	3-32
CEILING	3-32
CELLS.....	3-33
CELLTEXT	3-33
CFCMPXFVALUE	3-33
CFCMPXPVALUE	3-34
CFESCFVALUE	3-35
CFESCPVALUE	3-35
CFFESCINTEREST.....	3-36
CFFINITDEPOSIT	3-36
CFFPAYMENTS	3-37
CFFVALUE	3-37
CFFXCINTEREST	3-38
CFPESCINTEREST.....	3-38

CFPINITDEPOSIT	3-39
CFPINTEREST	3-39
CFPPAYMENTS	3-40
CFPVALUE	3-40
CFPXCTINTEREST	3-41
CFXCTFVALUE	3-41
CFXCTPVALUE	3-42
CHAR.....	3-42
CHIDIST	3-43
CHIINV	3-43
CHOOSE.....	3-44
CMONTH.....	3-45
CODE.....	3-45
COL.....	3-46
COLLATE.....	3-47
COLLATIONTABLE	3-48
COLLATIONTYPE	3-48
COLOF.....	3-49
COLS.....	3-50
COMBIN	3-50
COMPLEX.....	3-51
CONFIDENCE.....	3-51
CONTAINS	3-52
CONVERT	3-52
CORREL	3-55
COS	3-56
COSH	3-56
COT.....	3-57
COUNT	3-57
COUNTIF.....	3-57
COUPDAYBS.....	3-58
COUPDAYS	3-59
COUPDAYSNC	3-59
COUPNCD.....	3-60
COUPNUM.....	3-61
COUPPCD.....	3-61

COVAR.....	3-62
CRITBINOM.....	3-62
CTERM.....	3-63
CUMIPMT.....	3-63
CUMPRINC.....	3-64
CURRENCY.....	3-64
CUSTOMFORMAT.....	3-65
CUSTOMFORMATS.....	3-65
CWEEKDAY.....	3-66
CYAN.....	3-66
DATE.....	3-67
DATEVALUE.....	3-67
DAVG.....	3-68
DAY.....	3-69
DAYNAME.....	3-70
DB.....	3-70
DBTDISCOUNT.....	3-71
DBTDVALUE.....	3-71
DBTENDBAL.....	3-71
DBTINTEREST.....	3-72
DBTVALUE.....	3-72
DCOUNT.....	3-73
DDB.....	3-74
DEC2BIN.....	3-75
DEC2HEX.....	3-76
DEC2OCT.....	3-76
DECTOFRAC.....	3-77
DEGREES.....	3-77
DELTA.....	3-78
DEVSQ.....	3-78
DISC.....	3-78
DMAX.....	3-79
DMIN.....	3-80
DOLLARDE.....	3-81
DOLLARFR.....	3-82
DPDBDEPRECIATION.....	3-82

DPFYDEPRECIATION.....	3-83
DPSLDEPRECIATION	3-83
DPSMDEPRECIATION	3-84
DSTD.....	3-84
DSTDEV	3-85
DSUM	3-86
DSUMSQ	3-87
DTFPERIODS.....	3-89
DTPERIODS	3-89
DURATION	3-89
DVAR.....	3-90
DVARIANCE	3-91
E	3-93
EFFECT.....	3-93
ERF.....	3-93
ERFC.....	3-94
ERR	3-94
ERRNR.....	3-94
EVALUATE.....	3-95
EVEN	3-95
EXACT.....	3-96
EXP	3-96
EXPONDIST	3-97
EXPONENTIAL	3-97
EXPSMOOTH.....	3-98
FACTDOUBLE.....	3-98
FACTORIAL.....	3-99
FALSE.....	3-99
FDIST.....	3-100
FIND.....	3-100
FINV.....	3-101
FISHER	3-101
FISHERINV	3-102
FLOOR.....	3-102
FORECAST.....	3-102
FORMULAS	3-103

FORMULATEXT	3-103
FREQUENCY	3-104
FTEST	3-104
FV	3-105
FVDEPAMOUNT	3-106
FVDEPOSITS	3-106
FVDRATE.....	3-107
FVGROWTHRATE.....	3-107
FVGROWTHTIME.....	3-107
FVINTEREST	3-108
FVL	3-108
FVSCHEDULE.....	3-109
FVVALUE	3-109
GAMMADIST	3-110
GAMMAINV	3-110
GAMMALN.....	3-111
GCD	3-111
GEOMEAN	3-111
GESTEP	3-112
GOAL.....	3-112
GREEN.....	3-113
GREY	3-114
GROWTH	3-114
GUESS	3-114
HARMEAN.....	3-115
HEX2BIN	3-115
HEX2DEC.....	3-116
HEX2OCT.....	3-116
HLOOKUP.....	3-117
HOUR.....	3-118
HSBTORGB.....	3-119
HUE.....	3-119
HYPGEOMDIST	3-120
IF	3-121
IMABS	3-121
IMAGINARY.....	3-122

IMARGUMENT	3-122
IMCONJUGATE.....	3-122
IMCOS	3-123
IMDIV	3-123
IMEXP.....	3-123
IMLN.....	3-124
IMLOG10.....	3-124
IMLOG2.....	3-124
IMPOWER	3-125
IMPRODUCT	3-125
IMREAL.....	3-125
IMSIN.....	3-126
IMSQRT	3-126
IMSUB	3-126
IMSUM	3-127
INDEX.....	3-127
INDIRECT	3-128
INPUTCELL1	3-129
INPUTCELL2	3-129
INT	3-129
INTERCEPT	3-130
INTEREST	3-130
INTRATE.....	3-131
INVORDER	3-132
INVSAFETYSTOCK.....	3-132
INVSERVICELEVEL.....	3-133
IPMT	3-133
IRR	3-134
ISBLANK.....	3-135
ISERR.....	3-135
ISNA.....	3-136
ISNUMBER	3-136
ISRANGE.....	3-137
ISSTRING	3-137
KURT	3-138
LARGE.....	3-138

LCM	3-139
LEFT	3-139
LENGTH	3-140
LINEST	3-140
LN.....	3-141
LOANTERM.....	3-141
LOG.....	3-142
LOGEST.....	3-142
LOGINV.....	3-143
LOGN.....	3-143
LOGNORMDIST	3-144
LOWER.....	3-144
MAGENTA	3-145
MAKEBOOKCELL.....	3-145
MAKEBOOKRANGE.....	3-145
MAKECELL	3-146
MAKERANGE	3-147
MATCH.....	3-147
MAX.....	3-148
MDETERM	3-149
MDURATION.....	3-149
MEDIAN	3-150
MID	3-150
MIN	3-151
MINUTE	3-151
MINVERSE.....	3-152
MIRR.....	3-152
MMULT	3-153
MOD.....	3-153
MODE	3-154
MONTH	3-154
MONTHNAME.....	3-155
MROUND	3-155
MULTINOMIAL	3-156
N	3-156
NA	3-156

NAME	3-157
NEGBINOMDIST	3-157
NFORMAT	3-157
NOMINAL	3-159
NORMAL.....	3-159
NORMDIST	3-160
NORMINV	3-160
NORMSDIST	3-160
NORMSINV.....	3-161
NOW	3-161
NPER.....	3-161
NPV	3-162
NRDIMENSIONS	3-163
NRELEMENTS.....	3-163
NRSHEETNAMEDRANGES	3-163
NRSHEETS.....	3-164
OCT2BIN	3-164
OCT2DEC	3-164
OCT2HEX.....	3-165
ODD	3-165
ODDFPRICE.....	3-166
ODDLYIELD	3-166
PEARSON.....	3-167
PERCENTILE.....	3-167
PERCENTRANK.....	3-168
PERMUT.....	3-168
PFALPHABETA.....	3-169
PFAVGRETURN.....	3-169
PFCALLVALUE.....	3-169
PFCMPXINTRETURN.....	3-170
PFCMPXWGHTRETURN	3-170
PFINTRETURN.....	3-171
PFREGRESSION	3-171
PFRETURN.....	3-172
PFSTNDEVRETURN.....	3-172
PFWGHTRETURN.....	3-173

PFYIELD.....	3-173
PI	3-173
PMT.....	3-174
POISSON	3-174
POWER.....	3-175
PPMT.....	3-175
PRICE.....	3-175
PRICEDISC.....	3-176
PRICEMAT	3-177
PRINCIPAL	3-177
PROB.....	3-178
PROBZSCORE	3-178
PRODUCT	3-179
PROPER.....	3-179
PV	3-179
PVDEFPAYVALUE.....	3-180
PVDEFVALUE.....	3-180
PVDISCOUNTRATE	3-181
PVL	3-181
PVPAYAMOUNT	3-182
PVPAYMENTS	3-182
PVPRESENTVALUE	3-183
PVSCHEDULE.....	3-183
PVVALUE	3-184
QUARTILE	3-184
QUOTIENT	3-185
RADIANS	3-185
RAND.....	3-186
RANDBETWEEN.....	3-186
RANGE	3-186
RANGENAME	3-187
RANGENAMED.....	3-188
RANGENAMES	3-188
RANGEPART	3-188
RANK.....	3-189
RATE.....	3-189

RECEIVED	3-190
RED	3-190
REFERENCE	3-191
RELATEDBOOKS	3-191
REPEAT	3-192
REPLACE	3-192
RGB	3-193
RIGHT	3-193
ROLLOVERYEAR	3-194
ROMAN	3-194
ROUND	3-194
ROUNDDOWN	3-195
ROUNDUP	3-195
ROW	3-196
ROWOF	3-196
ROWS	3-197
RSQ	3-197
S	3-198
SAMERANGE	3-198
SATURATION	3-199
SECOND	3-199
SERIESSUM	3-200
SHEET	3-200
SHEETCELLS	3-200
SHEETFORMULAS	3-201
SHEETLIST	3-201
SHEETNAMEOF	3-202
SHEETRANGE	3-202
SHEETRANGENAMES	3-202
SHEETS	3-203
SIGN	3-203
SIN	3-204
SINH	3-204
SKEW	3-205
SLN	3-205
SLOPE	3-206

SMALL	3-206
SQRT.....	3-207
SQRTPI	3-207
STANDARDIZE	3-207
STD	3-208
STDEV	3-208
STDEVP.....	3-209
STEYX	3-209
STR.....	3-210
STRING.....	3-210
SUM	3-211
SUMIF.....	3-211
SUMPRODUCT.....	3-212
SUMSQ	3-212
SUMX2MY2	3-213
SUMX2PY2	3-213
SUMXMY2	3-214
SYD.....	3-214
TABLERANGE	3-215
TAN.....	3-215
TANH.....	3-216
TBILLEQ	3-216
TBILLPRICE	3-217
TBILLYIELD.....	3-217
TDIST.....	3-217
TERM.....	3-218
TICKER.....	3-218
TIME	3-219
TIMEVALUE.....	3-220
TINV	3-220
TITLESRANGE.....	3-221
TREND.....	3-221
TRIM.....	3-221
TRIMMEAN	3-222
TRUE.....	3-222
TRUNC	3-223

TTEST	3-223
UNIFORM.....	3-224
UPPER.....	3-224
VALUE	3-224
VAR	3-225
VARIANCE	3-225
VARP	3-226
VDB	3-226
VLOOKUP.....	3-227
WEIBULL	3-228
WHITE	3-229
XIRR	3-229
XNPV	3-229
YEAR.....	3-230
YELLOW	3-230
YIELD	3-231
YIELDDISC	3-231
YIELDMAT	3-232
ZSCOREPROB	3-232
ZTEST	3-233

CHAPTER 1: INTRODUCING FUNCTIONS

Functions are used in JSheet to perform data calculations, return range information, and return information on settings, selections, and status.

Functions are composed of a keyword that describes the function and a set of parentheses that contain any arguments needed by the function (SUM(expressions)). Some functions do not require any arguments (PI()).

Refer to *Chapter 3, Alphabetical List of Functions* for an alphabetical list of the JSheet functions.

Function Keywords

Keywords are required sections of functions that identify the function result. For example, ROW() returns the row number of the current cell in the worksheet. NOW() returns a serial date number that represents the current date and time. Some functions, as demonstrated by these examples, do not require arguments. Although keywords are not case-sensitive, they are displayed in uppercase in this manual.

Function Arguments

Arguments are divided into the following categories: number, string, range, and coordinate. An argument may be comprised of one or more

Chapter 1: Introducing Functions

numbers, strings, or ranges. When multiple arguments are used, the arguments are separated with commas.

Important: The comma is the default argument separator. If you change the separator, enter the character you supply as the string between multiple command arguments.

The following table describes the types of arguments in a command or function.

Component	Description
Number	Numeric data on which the function performs a calculation.
String	Textual data on which the function performs a calculation.
Range	References to a range of worksheet cells that contain data used in the function calculation.

Using Number Arguments

A number argument can consist of a number, or a function that returns a number. The following are examples of number arguments.

Examples

```
EXPONENTIAL(10)  
ROUND(COS(1.24356), 3)
```

Using Dates and Times as Arguments

Dates and times are also considered numbers. A special type of number, called a serial date number, is used to represent dates and times. The serial date number is the number of days since December 30, 1899, and can be any number from 0 to 401768 (December 31, 2999). A negative serial date number is evaluated as a date prior to December 30, 1899.

Times are represented by fractional portions of serial date numbers. For example, 0 represents midnight, .5 represents noon, .75 represents 6:00 p.m., and .875 represents 9:00 p.m. Therefore, the date number 34075.65 represents 3:36 p.m. on April 16, 1993.

The following example illustrates how to use a serial date number as a command and function argument.

Example

```
DAYNAME(38055)
```

Using String Arguments

A string argument is one or more alphabetic, numeric, or "special" characters (for example, !, @, _, or ~) contained in a set of double quotation marks. Numbers that are surrounded by a set of double quotation marks are evaluated as strings. Examples of string arguments are shown below.

Important: Always surround string arguments with a set of double quotation marks.

Examples

```
LENGTH("Hello World!")  
UPPER("This concludes the statistical evaluation of the study.")
```

Using Range Arguments

A range is a collection of worksheet cells. A range can consist of a single cell (such as, A1, B5, or C9) or a group of cells. If a range is a group of cells, the first cell is specified, followed by two periods, and then the ending cell reference (for example, A1..B5 or R1C1..R5C2).

Types of Worksheet Ranges

Worksheet ranges can be relative (for example, C2 and A1..D4), absolute (for example, \$C\$2 and \$A\$1..\$D\$4), or combined (for example, C\$2 and \$A1..\$D4). If a formula that contains a relative range is cut or copied and pasted, the range is modified to preserve the cell relationships. If a formula that contains an absolute range is cut or copied and pasted, the range does not change. You can refer to a range using any one of these reference options.

When you type a range name in a function, any names that contain spaces must be enclosed in a set of single quotation marks (for example, 'January Sales').

The following example contains range arguments.

Example

```
ISBLANK(B14)
```

Using Linked Ranges as Arguments

You can reference a cell or range on an active sheet that is not the current sheet by preceding the range reference or range name with the name of the noncurrent, active sheet and an exclamation mark (for example, Sheet2!A1..D5).

Important: If the name of a range contains a space, the entire range must be surrounded by a set of single quotation marks (for example, SUM('Sheet2!Monthly Sales')).

Using Optional Function Arguments

Some function arguments are optional. Optional function arguments are surrounded by a set of square brackets in this manual. The use of optional function arguments is dependent on the items you have selected. If you query information about a selected item, the optional argument is not necessary. If you query information about an unselected item, the optional argument is required to tell the function which item to query.

Using Formulas as Arguments

A formula is composed of one or more functions, numbers, strings, ranges, measurements, or operators, or any combination of these components. A formula performs a calculation on data and can be supplied as a function argument. The data in the formula is evaluated in the defined order of precedence.

The number, string, and range arguments that can be used in a formula are detailed in the previous sections of this chapter. This section explains the use of operators and the evaluation order of operators used in a formula.

Using Operators in a Formula

The operator that is used determines the type of calculation or evaluation that is performed. The operator also determines the numbers, ranges, strings, or measurements that are required by the formula.

The available operators are described in the following table.

Chapter 1: Introducing Functions

Operator Type	Definition
Numeric Operators. These operators return a number	These operators perform arithmetic operations on numbers. The numeric operators are: Addition (+) Subtraction (-) Multiplication (*) Division (/) Exponentiation (^)
Text Operator. This operator returns a string.	The text operator is an ampersand (&), which concatenates strings.
Relational Operators. These operators return a number (for example, 1 or 0).	These operators evaluate two numbers, ranges, or strings to determine whether they meet the criteria of the given operator. If the criteria are met, 1 (TRUE) is returned. If the criteria are not met, 0 (FALSE) is returned. The relational operators are: Equal to (=) Greater than (>) Greater than or equal to (>=) Less than (<) Less than or equal to (<=) Not equal to (<>)
Logical Operators. These operators return a number (for example, 1 or 0.)	These operators evaluate the formula and return a 1 if the evaluation is true and 0 if the evaluation is false. They are often used in conjunction with the relational operators. The logical operators are: AND Requires values on both sides of the operator. Returns 1 only if both values are true. OR Requires values on both sides of the operator. Returns 1 if either value is true. NOT Requires only one value on the right side of the operator. Returns 1 when the comparison is false.

The following examples demonstrate the use of the various formula operators in functions.

Examples

`ADDMONTHS(NOW(), 3 * 12)`

This `ADDMONTHS` function multiplies 3 by 12 and adds the total to the current date, which is returned by the `NOW` function.

Chapter 1: Introducing Functions

UPPER("First" & " Place")

This function concatenates the strings and changes the character to uppercase. The result of this function is FIRST PLACE.

IF(A32 > 45000, "Successful", "Unsuccessful")

This IF function also checks to see if the contents of worksheet cell A32 is greater than 45000. If the comparison is true, Successful is returned. Unsuccessful is returned if the comparison is not true.

IF(A32 > 45000 OR B32 = 28000, "Successful", "Unsuccessful").

This IF function checks to see if the contents of worksheet cell A32 is greater than 45000 or if cell B32 contains 28000. If the comparison is true, Successful is returned. Unsuccessful is returned if the comparison is not true.

The Evaluation Order in Formulas

When you use more than one component in a formula, the components are evaluated and calculated in a specific order of precedence. The order of precedence is explained in the following table.

Order of Precedence	Description
1 Parentheses	Items surrounded by parentheses receive highest precedence. If parentheses are nested within other parentheses, the innermost level receives the highest precedence.
2 Measurement operators	Numbers with measurement operators (INCH(ES), MIL(S), POINT(S), DECIPOINT(S), MILLIMETER(S)) are converted to twips.
3 Exponentiation	Exponential numbers are calculated.
4 Negative/Positive	The negative or positive state of numbers is considered.
5 Multiplication/Division	Numeric operations involving multiplication (*) or division (/) are calculated.

Chapter 1: Introducing Functions

6 Addition/Subtraction	Numeric operations involving addition (+) or subtraction (-) are calculated.
7 Relational Evaluation	Evaluations involving relational operators (=, >, >=, <, <=, <>) are performed.
8 Unary Logical	Evaluations involving the logical operator NOT are performed.
9 String or Logical	String concatenation (&) or logical evaluation involving the logical operators AND and OR are performed.

It is important to understand the order of evaluation when combining components. The order of precedence can easily change the order of formula evaluation, and ultimately, a formula's result.

CHAPTER 2: FUNCTIONS BY GROUP

The JSheet functions have been divided into twelve categories to help you identify and locate particular functions or function groups. The function categories are:

Numeric	Logical	Color
Financial	Database	Worksheet
Statistical	Text	Worksheet Status
Engineering	Date and Time	Environment

Numeric Functions

Mathematical calculations and numerical data analyses are performed with numeric functions. These functions can be entered directly into a worksheet cell. Numeric functions perform the following tasks:

Calculation Type	Description
Trigonometric Calculations	Trigonometric functions perform standard trigonometric operations.
Logarithmic Calculations	Logarithmic functions calculate common logarithms, logarithms according to the given base, and values based on the logarithmic constant.
Number Generation	Number generating functions produce random numbers from uniform, exponential, and normal distributions.

Chapter 2: Functions by Group

Numerical Analysis	Numerical analysis functions query a number to determine the number's absolute value, factorial, sign, and square root. Numerical analysis functions also determine integer values, round numbers to a specified precision, and return the remainder from a division operation.
--------------------	---

ABS()	EXP()	MROUND()	SIN()
ACOS()	EXPONENTIAL()	MULTINOMIAL()	SINH()
ACOSH()	FACTDOUBLE()	NORMAL()	SQRT()
ASIN()	FACTORIAL()	ODD()	SQRTPI()
ASINH()	FLOOR()	PI()	SUM()
ATAN()	GCD()	POWER()	SUMIF()
ATAN2()	GOAL()	PRODUCT()	SUMPRODUCT()
ATANH()	GUESS()	QUOTIENT()	SUMSQ()
CEILING()	INTO()	RADIANS()	SUMX2MY2()
COMBIN()	LCM()	RAND()	SUMX2PY2()
COS()	LN()	RANDBETWEEN()	SUMXMY2()
COSH()	LOG()	ROMAN()	TAN()
COT()	LOGN()	ROUND()	TANH()
COUNTIF()	MDETERM()	ROUNDDOWN()	TRUNC()
DEGREES()	MINVERSE()	ROUNDUP()	UNIFORM()
E()	MMULT()	SERIESSUM()	
EVEN()	MOD()	SIGN()	

Financial Functions

Financial functions analyze and compute accounting and financial data. These functions can be entered directly into a worksheet cell. The tasks performed by the financial functions are shown in the following table.

Calculation Type	Description
Loan calculations	Loan functions calculate interest rates, principals, and terms.
Bond calculations	Bond functions calculate bond prices, time to maturity, and yields.
Depreciation calculations	Depreciation functions calculate the depreciation value of an item based on the item value, salvage value, and life of the item.
Cash flow calculations	Cash flow functions calculate rates of incoming and outgoing cash flow, present values, and future values.

Chapter 2: Functions by Group

ACCRINT()	DBTDISCOUNT()	ODDLPRICE()
ACCRINTM()	DBTDVALUE()	ODDLYIELD()
AMORDEGRC()	DBTENDBAL()	PFALPHABETA()
AMORLINC()	DBTINTEREST()	PFVGRRETURN()
BNDCMPXPRICE()	DBTVALUE()	PFCALLVALUE()
BNDCMPXYIELD()	DDB()	PFCMPXINTRETURN()
BNDCONVEXITY()	DECTOFRAC()	PFCMPXWGHTRETURN()
BNDDURATION()	DISC()	PFINTRETURN()
BNDESTPRICE()	DOLLARDE()	PFREGRESSION()
BNDPRICE()	DOLLARFR()	PFRETURN()
BNDPRICECHANGE()	DPDBDEPRECIATION()	PFSTNDEVRETURN()
BNDSSENSITIVITY()	DPFYDEPRECIATION()	PFWGHTRETURN()
BNDYIELD()	DPSLDEPRECIATION()	PFYIELD()
BONDPRICE()	DPSMDEPRECIATION()	PMT()
BONDYTM()	DTFPERIODS()	PPMT()
CFCMPXFVALUE()	DTPERIODS()	PRICE()
CFCMPXPVALUE()	DURATION()	PRICEDISC()
CFESCFVALUE()	EFFECT()	PRICEMAT()
CFESCPVALUE()	FV()	PRINCIPAL()
CFFESCINTEREST()	FVDEPAMOUNT()	PV()
CFFINITDEPOSIT()	FVDEPOSITS()	PVDEFPAYVALUE()
CFFINTEREST()	FVDRATE()	PVDEFVALUE()
CFFPAYMENTS()	FVGROWTHRATE()	PVDISCOUNTRATE()
CFFVALUE()	FVGROWTHTIME()	PVL()
CFFXCTINTEREST()	FVINTEREST()	PVPAYAMOUNT()
CFPESCINTEREST()	FVL()	PVPAYMENTS()
CFPINITDEPOSIT()	FVSCHEDULE()	PVPRESENTVALUE()
CFPINTEREST()	FVVALUE()	PVSCHEDULE()
CFPPAYMENTS()	INTEREST()	PVVALUE()
CFPVALUE()	INTRATE()	RATE()
CFPXCTINTEREST()	INVORDER()	RECEIVED()
CFXCTFVALUE()	INVSAFETYSTOCK()	SLN()
CFXCTPVALUE()	INVSERVICELEVEL()	SYD()
COUPDAYBS()	IPMT()	TBILLEQ()
COUPDAYS()	IRR()	TBILLPRICE()
COUPDAYSNC()	LOANTERM()	TBILLYIELD()
COUPNCD()	MDURATION()	TERM()
COUPNUM()	MIRR()	VDB()
COUPPCD()	NOMINAL()	XIRR()
CTERM()	NPER()	XNPV()
CUMIPMT()	NPV()	YIELD()
CUMPRINC()	ODDFPRICE()	YIELDDISC()
DB()	ODDFYIELD()	YIELDMAT()

Statistical Functions

Statistical functions perform statistical calculations and analyses. These functions can be entered directly into a worksheet cell. The tasks that statistical functions perform are shown in the following table.

Calculation Type	Description
Counting	Counting functions find the total number, the minimum value, and the maximum value of specified items.
Dispersion Calculations	Dispersion functions calculate standard deviations and variances.
Numerical Calculations	Numerical functions sum, average, and square numbers.

AVEDEV()	FISHERINV()	MAX()	SLOPE()
AVG()	FORECAST()	MEDIAN()	SMALL()
BETADIST()	FREQUENCY()	MIN()	STANDARDIZE()
BETAINV()	FTEST()	MODE()	STD()
BINOMDIST()	GAMMADIST()	NEGBINOMDIST()	STDEV()
CHIDIST()	GAMMAINV()	NORMSDIST()	STDEVP()
CHIINV()	GAMMALN()	NORMSINV()	STEYX()
CHITEST()	GEOMEAN()	PEARSON()	TDIST()
CONFIDENCE()	GROWTH()	PERCENTILE()	TINV()
CORREL()	HARMEAN()	PERCENTRANK()	TREND()
COUNT()	HYPGEOMDIST()	PERMUT()	TRIMMEAN()
COVAR()	INTERCEPT()	POISSON()	TTEST()
CRITBINOM()	KURT()	PROB()	VAR()
DEVSQ()	LARGE()	PROBZSCORE()	VARIANCE()
EXPONDIST()	LINEST()	QUARTILE()	VARP()
EXPSMOOTH()	LOGEST()	RANK()	WEIBULL()
FDIST()	LOGINV()	RSQ()	ZSCOREPROB()
FINV()	LOGNORMDIST()	SKEW()	ZTEST()
FISHER()			

Engineering Functions

Engineering functions convert and distribute real and imaginary numbers, test return values of numbers, return the difference of complex numbers, and more.

BESSELI()	CONVERT()	IMABS()	IMPRODUCT()
BESSELJ()	DEC2BIN()	IMAGINARY()	IMREAL()
BESSELK()	DEC2HEX()	IMARGUMENT()	IMSIN()
BESSELY()	DEC2OCT()	IMCONJUGATE()	IMSQRT()
BIN2DEC()	DELTA()	IMCOS()	IMSUB()
BIN2HEX()	ERF()	IMDIV()	IMSUM()
BIN2OCT()	ERFC()	IMEXP()	OCT2BIN()
BITAND()	GESTEP()	IMLN()	OCT2DEC()
BITOR()	HEX2BIN()	IMLOG10()	OCT2HEX()
BITXOR()	HEX2DEC()	IMLOG2()	
COMPLEX()	HEX2OCT()	IMPOWER()	

Logical Functions

Logical functions query data to determine if the data meets the condition specified by the function. Most logical functions have a predefined condition for which they query. These functions return 0 if the condition is false and 1 if the condition is true. For example, use the IF function in a worksheet cell or script to specify a condition and its possible return values.

CHOOSE()	ISNA()
EVALUATE()	ISNUMBER()
FALSE()	ISRANGE()
IF()	ISSTRING()
ISBLANK()	SAMERANGE()
ISERR()	TRUE()

Database Functions

Database functions query information from worksheet ranges based on the criteria you specify. A database is defined in a worksheet by using columns to represent fields and rows to represent database records. All database functions require a database argument, offset, and criterion. These functions can be entered directly into a worksheet cell.

The database argument is a range reference or range name that defines the dimensions of the database. The column offset argument is a number that indicates which column is used for the formula calculation. The offset begins with the first column of the database range being 0 and increments by 1 for each additional column to the right. The criteria argument is used to define which columns in the worksheet are included in the formula calculation.

The tasks performed by the database functions are shown in the following table.

Calculation Type	Description
Counting	Counting functions find the total number of items, the minimum value, and the maximum value.
Dispersion Calculations	Dispersion functions calculate standard deviations and variances.
Numerical Calculations	Numerical functions sum, average, and square numerical data.

DAVG()	DSTDEV()
DCOUNT()	DSUM()
DMAX()	DSUMSQ()
DMIN()	DVAR()
DSTD()	DVARIANCE()

Text Functions

Text functions locate, compare, and manipulate strings. These functions can be entered directly into a worksheet cell. The following table describes the tasks text functions perform.

Calculation Type	Description
Searching	Text search functions locate strings of text according to the parameters specified by the function.
Conversion	Conversion functions change the case of strings, convert strings to numbers, and determine ASCII character equivalents.
Comparisons	Comparison functions evaluate strings to determine likeness and sorting order.

CHAR()	MID()
CODE()	NFORMAT()
COLLATE()	PROPER()
CONTAINS()	REPEAT()
CURRENCY()	REPLACE()
EXACT()	RIGHT()
FIND()	STR()
FORMULATEXT()	STRING()
LEFT()	TRIM()
LENGTH()	UPPER()
LOWER()	VALUE()
MATCH()	

Date and Time Functions

Date and time functions perform calculations and conversions on serial date numbers and strings. These functions can be entered directly into a worksheet cell. The following table shows the tasks performed by the date and time functions.

A serial date number is number (from 0 to 401768) that indicates the number of days since December 30, 1899.

Chapter 2: Functions by Group

Calculation Type	Description
Adding and Subtracting	These functions add or subtract days, hours, minutes, months, seconds, and years to or from the given serial date number.
Conversions	Conversion functions change a date or time string to a serial date number. Conversion functions also change a serial date number to a string.

ADATE()	CMONTH()	MONTH()
ADDDAYS()	CWEEKDAY()	MONTHNAME()
ADDDHOURS()	DATE()	NOW()
ADDMINUTES()	DATEVALUE()	SECOND()
ADDMONTHS()	DAY()	TIME()
ADDSECONDS()	DAYNAME()	TIMEVALUE()
ADDYEARS()	HOUR()	YEAR()
ATIME()	MINUTE()	

Color Functions

The color functions return the RGB color value of a particular color or color component.

BLACK()	GREEN()	MAGENTA()	WHITE()
BLUE()	GREY()	RED()	YELLOW()
BRIGHTNESS()	HSBTORGB()	RGB()	
CYAN()	HUE()	SATURATION()	

Worksheet Functions

Worksheet functions provide a mechanism for querying and retrieving information from a worksheet and for specifying range references. These functions can be entered directly into a worksheet cell. The tasks these functions perform are displayed in the following table.

Chapter 2: Functions by Group

Calculation Type	Description
Worksheet Reference	Worksheet reference functions enable you to reference a cell or range in terms other than the standard cell or range reference (for example, A1, R1C1, A1..D4, or "A1..D1").
Location	Location functions return the currently selected cell, range, column, or row.
Data query	Data query functions return the contents of the selected cell or range.

CELL()	INDIRECT()	RANGEPART()
CELLTEXT()	MAKEBOOKCELL()	ROW()
COL()	MAKEBOOKRANGE()	ROWOF()
COLOF()	MAKECELL()	ROWS()
COLS()	MAKERANGE()	S()
COMMALIST()	N()	SHEETNAMEOF()
ERR()	NA()	SHEETS()
ERRNR()	NRDIMENSIONS()	TICKER()
HLOOKUP()	NRELEMENTS()	VLOOKUP()
INDEX()	RANGE()	

Worksheet Status Functions

These functions return information about the current worksheet. Use worksheet status functions in your scripts to verify that certain worksheet attributes and settings are applied.

BOOKOF()	NRSHEETS()	SHEETRANGE()
CELLS()	NRSORTKEYS()	SHEETRANGENAMES()
CRITERIARANGE()	RANGENAME()	SORTKEYS()
CUSTOMFORMAT()	RANGENAMED()	SORTRANGE()
CUSTOMFORMATS()	RANGENAMES()	TABLERANGE()
DATARANGE()	REFERENCE()	TITLESRANGE()
FORMULAS()	RELATEDBOOKS()	VALIDCUSTOMFORMAT()
INPUTCELL1()	SHEET()	VALIDRANGENAME()
INPUTCELL2()	SHEETCELLS()	VALUERANGE()
NAME()	SHEETFORMULAS()	
NRSHEETNAMEDRANGES()	SHEETLIST()	

Chapter 2: Functions by Group

CHAPTER 3: ALPHABETICAL LIST OF FUNCTIONS

This chapter contains an alphabetic listing of the JSheet functions. Syntax and return values are presented for each function. Examples are provided for clarity. Unless otherwise noted, the results of these examples are rounded to two decimal places.

ABS

Syntax: ABS(number)

Returns: Number

Definition: Use ABS to return the absolute value of the specified number. An absolute value is the positive form of a number.

Example	Result
ABS(9.20)	9.20
ABS(0)	0.00
ABS(-12.00)	12.00

The first two examples return the original number, demonstrating that positive numbers and zero values are absolute. The third example returns the absolute value of a negative number, which is its positive counterpart.

Related Function: SIGN.

Chapter 3: Alphabetical List of Functions

ACCRINT

Syntax: ACCRINT(issue, first_interest, settlement, rate, par, frequency, basis)

Returns: Number

Definition: This function returns the accrued interest for a security that pays periodic interest. Issue, first_interest, and settlement are relevant dates. Rate is the coupon rate. Par is the par value. Frequency is the number of payments. Basis is the day count basis.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

ACCRINT(date(93,2,28),date(93,8,31),date(93,5,1), 0.1, 1000, 2,0)

Result

16.94444

ACCRINTM

Syntax: ACCRINTM(issue, maturity, rate, par, basis)

Returns: Number

Definition: This function returns the accrued interest for a security that pays interest at maturity. Issue and maturity are relevant dates. Rate is the coupon rate. Par is the par value. Basis is the day count basis.

Chapter 3: Alphabetical List of Functions

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example	Result
ACCRINTM(DATE(93,4,1),DATE(93,6,15), 0.1, 1000, 3)	20.54795

ACOS

Syntax: ACOS(number)

Returns: Number in radians ranging from 0 to pi

Definition: Use ACOS to return the arc cosine of the specified number. Number can be a number from -1 to 1.

Example	Result
ACOS(0.75)	0.72
1 / ACOS(0.75)	1.38
ACOS(COS(2))	2.00

The first example returns the arc cosine of .75. The second example divides 1 by the arc cosine of .75. The third example calculates the arc cosine of the cosine of 2.

Related Functions: ACOSH, ASIN, ASINH, ATAN, ATAN2, ATANH, COS, COSH, SIN, SINH, TAN, and TANH.

Chapter 3: Alphabetical List of Functions

ACOSH

Syntax: ACOSH(number)

Returns: Number

Definition: Use ACOSH to calculate the hyperbolic arc cosine of a number greater than or equal to 1.

Example	Result
ACOSH(2)	1.32
1/ACOSH(10)	0.33

The first example calculates the hyperbolic arc cosine of 2. The second example divides 1 by the hyperbolic arc cosine of 10.

Related Functions: ACOS, ASIN, ASINH, ATAN, ATAN2, ATANH, COS, COSH, SIN, SINH, TAN, and TANH.

ADATE

Syntax: ADATE(date_number, picture)

Returns: String

Definition: Use ADATE to convert a serial date number (date_number) to a string using the specified picture format. (A serial date number is a number from 0 to 401768 that indicates the number of days since December 30, 1899.) Picture defines the format used to create the date string. The following table defines the available picture elements.

Chapter 3: Alphabetical List of Functions

Element	Description
d	A one- or two-digit day (for example, 1 or 30)
dd	A two-digit day (for example, 01.)
day	The full day name (for example, Saturday).
m	A one- or two-digit month (for example, 1 or 12).
mm	A two-digit month (for example, 01 or 12).
mon	An abbreviated month name (for example, Jan.)
month	The full month name (for example, January)
y	A one- or two-digit year (for example, 1 for the year 2001 and 98 for the year 1998).
yy	A two-digit year (for example, 01 for the year 2001 and 98 for the year 1998.)
yyy	A four-digit year (for example, 2001 or 1998).

Example

ADATE(32328, "month d, yyyy")

ADATE(38233, "dd-mm-yyyy")

ADATE(DATEVALUE("03/24/92"), "mon d yy")

Result

July 4, 1988

03-09-2004

Mar 24 92

In the first and second examples, ADATE returns a formatted serial date number using the specified picture format.

In the third example, DATEVALUE determines the number of days from December 30, 1899, to March 24, 1992 (33687 days). The result is the argument for the ADATE function that returns the date as "mon d yy."

Related Functions: ADDDAYS, ATIME, DATE, DATEVALUE, FORMAT, NFORMAT, and NOW.

ADDDAYS

Syntax: ADDDAYS(date_number, days)

Returns: Serial date number

Chapter 3: Alphabetical List of Functions

Definition: Use `ADDDAYS` to add the number of days to the given serial date number (`date_number`). (A serial date number is a number from 0 to 401768 that indicates the number of days since December 30, 1899.) If days is negative, the days are subtracted from `date_number`.

Example	Result
<code>ADDDAYS(32330, 5)</code>	32335
<code>ADDDAYS(32330, -10)</code>	32320
<code>ADDDAYS(38233, SUM(22, 31, 18))</code>	38304

Five days are added to `date_number` in the first example. In the second example days is negative, so 10 days are subtracted from `date_number`. In the third example, `SUM` determines the number of days (71) to add to 38233.

Related Functions: `ADATE`, `ADDMONTHS`, `ADDYEARS`, `DAY`, `DAYNAME`, and `FORMAT`.

ADDDAYS

Syntax: `ADDDAYS(date_number, hours)`

Returns: Serial date number

Definition: Use `ADDDAYS` to add the number of days to the given serial date number (`date_number`). (A serial date number is a number from 0 to 401768 that indicates the number of days since December 30, 1899. The decimal portion of the number is a fraction of one day.) If hours is negative, the hours are subtracted from `date_number`.

Example	Result
<code>ADDDAYS(33436.25, 10)</code>	33436.67
<code>ADDDAYS(32328.25, -3)</code>	32328.13

Chapter 3: Alphabetical List of Functions

The first example adds 10 hours to the serial date number 33436.25. The second example subtracts three hours from the serial date number 32328.25.

Related Functions: ADMINUTES, ADDSECONDS, and HOUR.

ADMINUTES

Syntax: ADMINUTES(date_number, minutes)

Returns: Serial date number

Definition: Use ADMINUTES to add the number of minutes to the given serial date number (date_number). If minutes is negative, the minutes are subtracted from date_number.

Example	Result
ADMINUTES(32328.375, 90)	32328.4375
ADMINUTES(38231.5, -15 * 60)	38230.875

The first example adds 90 minutes to the serial date number 32328.375. The second example subtracts 900 minutes (15*60) from the serial date number 38231.5.

Related Functions: ADDHOURS and ADDSECONDS.

ADDMONTHS

Syntax: ADDMONTHS(date_number, months)

Returns: Serial date number

Definition: Use ADDMONTHS to add the months specified to the date_number. If months is negative, it is subtracted from date_number.

Chapter 3: Alphabetical List of Functions

Example	Result
ADDMONTHS(32328, 1)	32359
ADDMONTHS(32328, -14)	31901
ADDMONTHS(DATEVALUE("09/03/ 2004"), 5)	38386

The first example adds one month to 32328. The second example subtracts 14 months from 32328. The last example uses DATEVALUE to convert 09/03/2004 to the serial date number 38233. ADDMONTHS then adds five months to this number.

Related Functions: ADDDAYS and ADDYEARS.

ADDSECONDS

Syntax: ADDSECONDS(date_number, seconds)

Returns: Serial date number

Definition: Use ADDSECONDS to add the number of seconds to date_number. If seconds is negative, it is subtracted from date_number.

Example	Result
ADDSECONDS(32328.75, 1800)	32328.771
ADDSECONDS(32328.75, - (60 * 60 * 6))	32328.500

The first example adds 1800 seconds to the serial date number 32328.75. The second example subtracts 21600 seconds (60*60*6) from the serial date number 32328.75.

Related Functions: ADDHOURS and ADDSECONDS.

Chapter 3: Alphabetical List of Functions

ADDEYEARS

Syntax: ADDEYEARS(date_number, years)

Returns: Serial date number

Definition: Use ADDEYEARS to add the number of years to date_number. If years is negative, it is subtracted from date_number.

Example	Result
ADDEYEARS(32328, 3)	33423
ADDEYEARS(32328, -7)	29771
ADDEYEARS(DATEVALUE("Sep 3, 2004"), 17)	44442

The first example adds three years to the serial date number 32328. The second example subtracts seven years from 32328. In the third example, DATEVALUE is used to convert Sept 3, 2004, into the serial date number 38233. Adding 17 years to this number results in a new serial date number of 44442 (September 3, 2021).

Related Functions: ADDEAYS and ADDMONTHS.

AMORDEGRC

Syntax: AMORDEGRC(cost, purchased, first, salvage, period, rate, basis)

Returns: Number

Definition: This function returns the depreciation for each period. This function is provided for the French accounting system. If an asset is purchased in the middle of the accounting period, then the prorated depreciate is taken into account. The function is similar to AMORLINC, except that a depreciation coefficient is applied in the calculation

Chapter 3: Alphabetical List of Functions

depending on the life of the assets. Cost is the asset cost. Purchased is the purchase date. First is the end date of the first period. Salvage is the salvage value at the end of the life of the asset. Period is the period. Rate is the depreciation rate. Basis is the year_base to be used.

Basis	Day System
0	360 days (NASD method)
1	Actual
3	365 days in a year
4	360 days in a year (European method)

Example	Result
AMORDEGRC(2400,34199,34334,300,1,0.15,1)	\$775

AMORLINC

Syntax: AMORLINC(cost, purchased, first, salvage, period, rate, basis)

Returns: Number

Definition: This function returns the depreciation for each accounting period. This function is provided for the French accounting system. If an asset is purchased in the middle of the accounting period, then the prorated depreciate is taken into account. Cost is the asset cost. Purchased is the purchase date of the asset. First is the end date of the first period. Salvage is the salvage value at the end of the life of the asset. Period is the period. Rate is the depreciation rate. Basis is the year_base to be used.

Basis	Day System
0	360 days (NASD method)
1	Actual
3	365 days in a year
4	360 days in a year (European method)

Example	Result
AMORLINC(2400,34199,34334,300,1,0.15,1)	\$360

Chapter 3: Alphabetical List of Functions

ASIN

Syntax: ASIN(number)

Returns: Number of radians, ranging from $\pi/2$ to $-\pi/2$

Definition: Use ASIN to return the arc sine of number, where number is the sine of the resulting angle. Number can be from 1 to -1 .

Example	Result
ASIN(0.17)	0.17
ASIN(-0.5)	-0.52
1 / ASIN(0.17)	5.85

The first two examples determine the arc sine of 0.17 and -0.5 , respectively. In the last example, the arc cosecant (5.85) is represented by the reciprocal of the arc sine of 0.17.

Related Functions: ACOS, ACOSH, ASINH, ATAN, ATAN2, ATANH, COS, COSH, PI, SIN, SINH, TAN, and TANH.

ASINH

Syntax: ASINH(number)

Returns: Number

Definition: ASINH returns the hyperbolic arc sine of number.

Example	Result
ASINH(1)	0.88
ASINH(0)	0.00

These examples calculate the hyperbolic arc sine of 1 and 0, respectively.

Related Functions: ACOS, ACOSH, ASIN, ATAN, ATAN2, ATANH, COS, COSH, SIN, SINH, TAN, and TANH.

Chapter 3: Alphabetical List of Functions

ATAN

Syntax: ATAN(number)

Returns: Number in radians, ranging from $\pi/2$ and $-\pi/2$

Definition: Use ATAN to return the arc tangent of number. ATAN is the inverse of TAN, which returns the tangent of a number in radians.

Example	Result
ATAN(1)	0.79
ATAN(0)	0.00

Related Functions: ACOS, ACOSH, ASIN, ASINH, ATAN2, ATANH, COS, COSH, PI, SIN, SINH, TAN, and TANH.

ATAN2

Syntax: ATAN2(x,y)

Returns: Number in radians, ranging from $\pi/2$ and $-\pi/2$

Definition: Use ATAN2 to return the angle between the positive x-axis and the line starting at the origin that passes through the specified coordinates x and y.

Example	Result
ATAN2(3, 15)	1.37
ATAN2(4, 5)	0.90
ATAN2(-1, 0)	3.14

The first example calculates the angle between the positive x-axis and a line that runs from (0, 0) through (3, 15). The second example determines the angle between the positive x-axis and a line that runs from (0, 0) through (4, 5). The last example returns 3.14 (π) as the angle between the positive axis and a line from (0, 0) through (-1, 0).

Related Functions: ACOS, ACOSH, ASIN, ASINH, ATAN, ATANH, COS, COSH, PI, SIN, SINH, TAN, and TANH.

ATANH

Syntax: ATANH(number)

Returns: Number

Definition: Use ATANH to return the hyperbolic arc tangent of the specified number. The argument may be any number less than 1 and greater than -1.

Example	Result
ATANH(0.5)	0.55
ATANH(-0.99)	-2.65

These examples calculate the hyperbolic arc tangent of 0.5 and -0.99, respectively.

Related Functions: ACOS, ACOSH, ASIN, ASINH, ATAN, ATAN2, COS, COSH, PI, SIN, SINH, TAN, and TANH.

ATIME

Syntax: ATIME(date_number, picture)

Returns: String

Definition: Use ATIME to convert the specified serial date number (date_number) to a string using the specified picture format. (A serial date number is a number from 0 to 401768 that indicates the number of days since December 30, 1899. The decimal portion of the number is a fraction of one day.) Picture defines the time format used to create the time string.

Chapter 3: Alphabetical List of Functions

The numbers to the right of the decimal point in `date_number` represent a decimal fraction of a day (for example, 0 = midnight, 0.5 = noon, and 0.75 = 6:00 PM). Time units can be separated by spaces, colons, or dashes.

The following table defines the valid picture elements.

Element	Description
hh	A number representing the hour in 24-hour format (for example, 16 is four o'clock in the afternoon).
hr	A number representing the hour in 12-hour format (for example, 04 is four o'clock).
mn	A number representing the minutes.
sc	A number representing the seconds.
hn	A number representing the hundredths of a second.
am	A two letter string indicating the time of day (for example, AM for times from midnight to noon and PM for times from noon to midnight).

Example	Result
<code>ATIME(32328.7, "hh:mn:sc")</code>	16:48:00
<code>ATIME(32328.7, "hr-mn-sc am")</code>	04-48-00 PM
<code>ATIME(TIMEVALUE("4:48 PM"), "hh-mn-sc")</code>	16-48-00

The first example returns the serial date number (32328.7) in the specified format. The second example returns the serial date number (32328.7), formatted with dashes, and AM or PM. The last example calculates the `TIMEVALUE` of 4:48 PM and displays the returned value in the specified format.

Related Functions: `ADATE`, `DATE`, `DATEVALUE`, `FORMAT`, `NFORMAT`, `NOW`, `TIME`, and `TIMEVALUE`.

AVEDEV

Syntax: `AVEDEV(number1, number2,...)`

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the average of the absolute deviations of data points from their mean. AVEDEV is a measure of the variability in a data set. Number1, number2,..... are 1 to 50 arguments for which you want the average of the absolute deviations.

Example	Result
AVEDEV(4,5,6,7,5,4,3)	1.020408

AVG

Syntax: AVG(expressions)

Returns: Number

Definition: Use AVG to return the average (arithmetic mean) of up to 50 entries in the specified list (expressions). Strings are evaluated as zero and blank cells are not included in the average.

This function is equivalent to SUM(expressions)/COUNT(expression).

Example	Result
AVG(13, 22, 17, 33, 8)	18.60
AVG(-45, 23, 0, 27, "88")	1.00
AVG(A1..A10)	5.50

The first example divides the sum of the expressions (93) by the number of items in the list (5). In the second example, the last argument in the list is a string ("88") and has a value of zero. Therefore, the sum (5) is divided by the number of items (5). In the last example, the range A1..A10 is filled with numbers from 1 to 10, incremented by 1. The average of 1, 2, 3, 4, 5, 6, 7, 8, 9, and 10 is 5.50.

Related Functions: COUNT, DAVG, DSUM, STDEV, and SUM.

Chapter 3: Alphabetical List of Functions

BESSELI

Syntax: BESSELI(x, n)

Returns: Number

Definition: This function returns the modified Bessel function, which is equivalent to the Bessel function evaluated for purely imaginary arguments. X is the value at which to evaluate the function. N is the order of the Bessel function.

$$I_n(x) = (i)^{-n} J_n(ix)$$

Example

BESSELI(1.5,1)

Result

0.981666

BESSELJ

Syntax: BESSELJ(x, n)

Returns: Number

Definition: This function returns the Bessel function. X is the value at which to evaluate the Bessel function. N is the order of the Bessel function.

$$J_n(x) = \sum_{k=0}^{\infty} \frac{(-1)^k}{k! \Gamma(n+k+1)} \left(\frac{x}{2}\right)^{n+2k}$$

where:

$$\Gamma(n+k+1) = \int_0^{\infty} e^{-x} x^{n+k} dx$$

is the Gamma function.

Example

BESSELJ(1.9,2)

Result

0.329926

BESSELK

Syntax: BESSELK(x, n)

Returns: Number

Definition: This function returns the modified Bessel function, which is equivalent to the Bessel function evaluated for purely imaginary arguments. X is the value at which to evaluate the function. N is the order of the function.

$$K_n(x) = \frac{P}{2} i^{n+1} [J_n(ix) + iY_n(ix)]$$

Example

BESSELK(1.5,1)

Result

0.277388

BESSELY

Syntax: BESSELY(x, n)

Returns: Number

Definition: This function returns the Bessel function, which is also called the Weber function or the Neumann. X is the value at which to evaluate the function. N is the order of the function.

$$Y_n(x) = \lim_{\nu \rightarrow n} \frac{J_\nu(x) \cos(\nu \pi) - J_{-\nu}(x)}{\sin(\nu \pi)}$$

where:

$$\text{ERF}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt$$

Example

BESSELY(2.5,1)

Result

0.145918

Chapter 3: Alphabetical List of Functions

BETADIST

Syntax: BETADIST(x, alpha, beta, A, B)

Returns: Number

Definition: This function returns the cumulative beta probability density function. The cumulative beta probability density function is commonly used to study variation in the percentage of something across samples, such as the fraction of the day people spend watching television. X is the value between A and B at which to evaluate the function. Alpha and beta are parameters to the distribution. A is an optional lower bound to the interval of x. B is an optional upper bound to the interval of x.

Example

BETADIST(2,8,10,1,3)

Result

0.685470581

BETAINV

Syntax: BETAINV(probability, alpha, beta, A, B)

Returns: Number

Definition: This function returns the inverse of the cumulative beta probability density function. That is, if $\text{probability} = \text{BETADIST}(x, \dots)$, then $\text{BETAINV}(\text{probability}, \dots) = x$. The cumulative beta distribution can be used in project planning to mode probable completion times given an expected completion time and variability. Probability is a probability associated with the beta distribution. Alpha and beta are parameters to the distribution. A is an optional lower bound to the interval of x. B is an optional upper bound to the interval of x.

Example

BETAINV(0.685470581,8,10,1,3)

Result

2

Chapter 3: Alphabetical List of Functions

BIN2DEC

Syntax: BIN2DEC(number)

Returns: Decimal number

Definition: This function converts a binary number to a decimal. Number is the binary number you want to convert. Binary numbers are limited to 10 digits.

Example	Result
BIN2DEC(1100100)	100
BIN2DEC(1111111111)	-1

Related Functions: BIN2HEX, BIN2OCT, DEC2BIN, DEC2HEX, DEC2OCT, HEX2BIN, HEX2DEC, HEX2OCT, OCT2BIN, OCT2DEC, and OCT2HEX.

BIN2HEX

Syntax: BIN2HEX(number, places)

Returns: Hexadecimal number as a string

Definition: This function converts a binary number to hexadecimal. Number is the binary number you want to convert. Places, which is optional, is the number of characters to use. If places is omitted, the function uses the minimum number of necessary characters.

Example	Result
BIN2HEX(11111011,4)	00FB
BIN2HEX(1110)	E
BIN2HEX(1111111111)	FFFFFFFF

Related Functions: BIN2DEC, BIN2OCT, DEC2BIN, DEC2HEX, DEC2OCT, HEX2BIN, HEX2DEC, HEX2OCT, OCT2BIN, OCT2DEC, and OCT2HEX.

Chapter 3: Alphabetical List of Functions

BIN2OCT

Syntax: BIN2OCT(number, places)

Returns: Octal number as a string

Definition: This function converts a binary number to octal. Number is the binary number you want to convert. Places, which is optional, is the number of characters to use. If places is omitted, the function uses the minimum number of necessary characters.

Example	Result
BIN2OCT(1001,3)	011
BIN2OCT(01100100)	144
BIN2OCT(1111111111)	777777777

Related Functions: BIN2DEC, BIN2HEX, DEC2BIN, DEC2HEX, DEC2OCT, HEX2BIN, HEX2DEC, HEX2OCT, OCT2BIN, OCT2DEC, and OCT2HEX.

BINOMDIST

Syntax: BINOMDIST(number_s, trials, probability_s, cumulative)

Returns: Number

Definition: This function returns the individual term binomial distribution probability. Use BINOMDIST in problems with a fixed number of tests or trials, when the outcomes of any trial are only success or failure, when trials are independent, and when the probability of success is constant throughout the experiment. Number_s is the number of successes in trials. Trials is the number of independent trials. Probability_s is the probability of success on each trial. Cumulative is a logical value that determines the form of the function. If cumulative is TRUE, then BINOMDIST returns the cumulative distribution function, which is the probability that there is, at most, number_s success; if FALSE, it returns

Chapter 3: Alphabetical List of Functions

the probability mass function, which is the probability that there are number_s successes.

Example	Result
<code>BINOMDIST(6,10,0.5,FALSE())</code>	0.205078

BITAND

Syntax: `BINAND(number1, number2)`

Returns: Number

Definition: Use BITAND to calculate the binary (base 2) "AND" of the given arguments, number1 and number2. Both arguments can be any number, logical value, or string. A string is considered to have a value of zero. A logical comparison returns a value of 1 (TRUE) or 0 (FALSE), depending on the result of the evaluation.

Arguments are viewed in their binary representation. Then, each digit of the binary representation is compared to determine if there is a 1 in the same digit of both numbers. If so, a 1 is placed in the same digit in the result number. Otherwise, a 0 is placed in the digit position. The value returned is displayed in the decimal (base 10) equivalent of the binary result.

Example	Result
<code>BITAND(15, 5)</code>	5

In the example above, the binary equivalent of 15 is 1111; the binary equivalent of 5 is 0101. When comparing the bits from right to left, the digit 1 occurs in both numbers as the first and third digit. This signifies the "AND" relationship between the two binary numbers.

The binary representation (0101) is displayed as the decimal number 5.

Chapter 3: Alphabetical List of Functions

Binary Representation	Decimal Representation
1111	15
0101	05
0101	05

In the following example, the binary representations of 85 and 94 are compared. When both binary numbers contain a 1 for a given digit, the result is 1, signifying the "AND" relationship. The result, therefore, has the digit 1 in the third, fifth, and seventh positions (from right to left).

Example	Result
BITAND(85, 94)	84

The binary result (1010100) is displayed as the decimal number 84.

Binary Representation	Decimal Representation
1010101	85
1011110	94
1010100	84

Related Functions: BITOR, BITXOR, and BORDERSTYLE.

BITOR

Syntax: BITOR(number1, number2)

Returns: Number

Definition: Use BITOR to calculate the binary (base 2) "OR" of the given arguments, number1 and number2. Both arguments can be any number, logical value, or string. A string is considered to have a value of zero. A logical comparison returns a value of 1 (TRUE) or 0 (FALSE), depending on the result of the evaluation.

Values are compared as binaries. Each digit of the binary number is compared to see if there is a 1 in either of the binary digit positions. If a 1 is found in one or both of the binary digit positions, a 1 is placed in the

Chapter 3: Alphabetical List of Functions

same digit in the result. Otherwise, a 0 is placed in the digit position. The value returned is displayed as the decimal (base 10) equivalent of the binary result.

In the following example, the binary equivalent of 2 is 0010; the binary equivalent of 4 is 0100. When comparing the bits from right to left, the digit 1 occurs in the second and third positions. This signifies the "OR" relationship between the two binary numbers.

Example	Result
BITOR(2, 4)	6

The binary representation (0110) is displayed as the decimal number 6.

Binary Representation	Decimal Representation
0010	2
0100	4
0110	6

In the following example, the binary representations of 2 and 7 are compared. When either binary number contains a 1 for a given digit, the result is 1; this signifies the "OR" relationship. The result, therefore, has the digit 1 in the first, second, and third positions.

Example	Result
BITOR(2, 7)	7

As shown by the binary representation, the binary numbers contain 1 in the last three digits. The resulting binary number contains a 1 in the same digit positions. The decimal equivalent of binary 0111 is 7.

Binary Representation	Decimal Representation
0010	2
0111	7
0111	7

Related Functions: BITAND and BITXOR.

Chapter 3: Alphabetical List of Functions

BITXOR

Syntax: BITXOR(number1, number2)

Returns: Number

Definition: Use BITXOR to calculate the binary (base 2) "EXCLUSIVE OR" of the given arguments, number1 and number2. Both arguments can be any number, logical value, or string. A string is considered to have a value of zero. A logical comparison returns a value of 1 (TRUE) or 0 (FALSE), depending on the result of the evaluation.

Values are compared as binary numbers in memory. Each digit of the binary number is compared to see if there is a 1 in either of the binary digit positions but not in both numbers at the same digit position. If so, a 1 is placed in the same position in the result string. Otherwise, a 0 is placed in the position. The value returned is the decimal equivalent of the result string.

In the following example, the digits of the two binary numbers (2 and 4) are compared, showing a 1 in position 2 of the first number and a 1 in position 3 of the second number (from right to left). The resulting binary number contains 1 in positions 2 and 3, which is the decimal number 6.

Example	Result
BITXOR(2, 4)	6

The binary result (0110) is displayed as the decimal number 6.

Binary Representation	Decimal Representation
0010	2
0100	4
0110	6

In the following example, the digits of the two binary numbers (2 and 6) are compared. Position 2 of both numbers contains a 1 and position 3 of

Chapter 3: Alphabetical List of Functions

the second number contains a 1 (from right to left). The resulting binary number has a 1 in position 3, which is the decimal number 4.

Example	Result
BITXOR(2, 6)	4

The binary result (0100) is displayed as the decimal number 4.

Binary Representation	Decimal Representation
0010	2
0110	6
0100	4

Related Functions: BITAND and BITOR.

BLACK

Syntax: BLACK()

Returns: 0

Definition: Use BLACK to return the RGB value of pure black.

Related Functions: BLUE, CYAN, GREEN, GREY, MAGENTA, RED, RGB, WHITE, and YELLOW.

BLUE

Syntax: BLUE()

Returns: 16711680

Definition: Use BLUE to return the RGB value of pure blue.

Related Functions: BLACK, CYAN, GREEN, GREY, MAGENTA, RED, RGB, WHITE, and YELLOW.

Chapter 3: Alphabetical List of Functions

BNDCMPXPRICE

Syntax: BNDCMPXPRICE(value, return, cpn_rate, cpn_period, mat_date, pur_date, cal, rates, chngdates, invtmnt)

Returns: Number

Definition: This function returns the price of a bond relative to a specified yield to maturity with an independent rate for reinvestment income. The price is calculated based on face value, desired return, coupon rate, coupon periods per year, maturity date, purchase date, calendar type (0 = 365 days/year, 1 = 360 days/year, 30 day months), two correlating ranges of reinvestment rates and change dates, and investment term (0 = duration of analysis, 1 = current period).

Example

	Result
BNDCMPXPRICE(1000, 0.11, 0.06, 2, DATEVALUE("6/15/2005"), DATEVALUE("9/9/1990"), 1, rate_arr, chng_arr, 1, 0)	58.73
BNDCMPXPRICE(1000, 0.11, 0.06, 2, DATEVALUE("6/15/1996"), DATEVALUE("9/9/1990"), 1, rate_arr, chng_arr, 1, 80)	67.23

Rate_arr is a range that contains a single value: 0.095. Chng_arr is a range that contains a single value: DATEVALUE("1/1/1990").

BNDCMPXYIELD

Syntax: BNDCMPXYIELD(value, price, cpn_rate, cpn_period, mat_date, pur_date, cal, rates, chngdates, invtmnt, sprice)

Returns: Number

Definition: This function calculates the yield to maturity for a bond when reinvested at a rate different from the standard yield to maturity. The return value is based on the face value, purchase price, coupon rate, coupon periods per year, maturity date, purchase date, calendar type (0 =

Chapter 3: Alphabetical List of Functions

365 days/year, 1 = 360 days/year, 30 day months), two correlating ranges of reinvestment rates and change dates, investment term (0=duration of analysis, 1=current period), and the expected sale.

Example	Result
BNDCMPXYIELD(1000, 58.75, 0.06, 2, DATEVALUE("6/15/1996"), DATEVALUE("9/9/1990"), 1, rate_arr, chng_arr, 1, 80)	13.49%
BNDCMPXYIELD(1000, 58.75, 0.06, 2, DATEVALUE("6/15/2005"), DATEVALUE("9/9/1990"), 1, rate_arr, chng_arr, 1, 0)	10.48%

Rate_arr is a range that contains a single value: 0.095. Chng_arr is a range that contains a single value: DATEVALUE("1/1/1990").

BNDCONVEXITY

Syntax: BNDCONVEXITY(yld, comp_period, cpn_rate, cpn_period, cpn_payments)

Returns: Number

Definition: This function returns the convexity of a bond (in years) based on yield to maturity, compounding periods per year, coupon rate, coupon periods per year, and number of remaining coupon payments.

Example	Result
BNDCONVEXITY(0.09, 2, 0.066, 2, 34)	61.43
BNDCONVEXITY(0.09, 2, 0.09, 2, 40)	65.67

BNDDURATION

Syntax: BNDDURATION(yld, comp_period, cpn_rate, cpn_period, cpn_payments)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the duration of a bond (in years) based on yield to maturity, compounding periods per year, coupon rate, coupon periods per year, and number of remaining coupon payments.

Example	Result
BNDDURATION(0.11, 2, 0.06, 2, 28)	8.48
BNDDURATION(0.11, 2, 0.12, 2, 48)	8.76

BNDESTPRICE

Syntax: BNDESTPRICE(value, return, comp_period, cpn_rate, cpn_period, cpn_payments)

Returns: Number

Definition: This function returns the estimated price of a bond relative to a specified yield to maturity. The value is calculated based on face value, expected return, compound periods per year, coupon rate, coupon periods per year, and number of remaining coupon payments.

Example	Result
BNDESTPRICE(1000, 0.11, 2, 0.06, 2, 28)	64.70

BNDPRICE

Syntax: BNDPRICE(value, return, comp_period, cpn_rate, cpn_period, mat_date, pur_date, cal)

Returns: Number

Definition: This function returns the price of a bond relative to a specified yield to maturity, and is based on face value, expected return, compounding periods per year, coupon rate, coupon periods per year, maturity date, purchase date, and calendar type (0 = 365 days/year, 1 = 360 days/year, 30 day months).

Chapter 3: Alphabetical List of Functions

Example	Result
BNDPRICE(1000, 0.1, 2, 0.06, 2, DATEVALUE("6/15/2005"), DATEVALUE("9/9/1990"), 0)	69.45
BNDPRICE(1000, 0.1, 2, 0.06, 2, DATEVALUE("6/15/2005"), DATEVALUE("9/9/1990"), 1)	69.46

BNDPRICECHANGE

Syntax: BNDPRICECHANGE(yld, comp_period, cpn_rate, cpn_period, yld_change, cpn_payments)

Returns: Number

Definition: This function returns the estimated percentage change in bond price for a given change in yield. The estimate is based on yield to maturity, compounding periods per year, coupon rate, coupon periods per year, change in yield, and number of remaining coupon payments.

Example	Result
BNDPRICECHANGE(0.06716, 2, 0.04, 2, 0.0125, 18)	-8.55%

BNDSSENSITIVITY

Syntax: BNDSSENSITIVITY(yld, comp_period, cpn_rate, yld_change, cpn_period, mat_date, pur_cate, cal)

Returns: Number

Definition: This function returns the sensitivity of a bond to interest rate changes based on yield to maturity, compounding periods per year, coupon rate, change in yield, coupon periods per year, maturity date, purchase date, and calendar type (0 = 365 days/year, 1 = 360 days/year, 30 day months).

Example	Result
BNDSSENSITIVITY(0.1, 2, 0.06, 0.1, 2, DATEVALUE("6/15/2005"), DATEVALUE("9/9/1990"), 360)	-8.03%

Chapter 3: Alphabetical List of Functions

BNDYIELD

Syntax: BNDYIELD(value, cpn_rate, price, comp_period, mat_date, pur_cate, cal)

Returns: Number

Definition: This function returns the yield to maturity for a bond based on redemption value, coupon rate, price, compounding periods per year, coupon periods per year, maturity date, purchase date, and calendar type (0 = 365 days/year, 1 = 360 days/year, 30 day months).

Example	Result
BNDYIELD(1000, 0.0875, 96.25, 365, 2, DATEVALUE("10/1/1995"), DATEVALUE("9/9/1990"), 1)	9.48%

BONDPRICE

Syntax: BONDPRICE(yld, face, rate, ytm, pmts_yr)

Returns: Number

Definition: Use BONDPRICE to calculate the price of a bond based on its yield (yld), face value (face), coupon rate (rate), number of years to maturity (ytm), and number of coupon payments per year (pmts_yr). Any number can represent yield, face, and rate. Ytm and pmts_yr can be any positive integer.

Example	Result
BONDPRICE(0.11, 1000, 0.09, 8, 2)	895.38

Related Function: BONDYTM.

BONDYTM

Syntax: BONDYTM(price, face, rate, ytm, pmts_yr)

Returns: Number

Definition: Use BONDYTM to calculate the yield to maturity for a bond based on its price (price), face value (face), coupon rate (rate), number of years to maturity (ytm), and number of coupon payments per year (pmts_yr). Any number can represent price, face, and rate. Ytm and pmts_yr can be any positive integer.

Example	Result
BONDYTM(895, 1000, 0.09, 8, 2)	.11

In this example, if a bond's listed price is \$895, its face value is \$1000, its coupon rate is 9 percent, its maturity date is 8 years from now, and its coupon payments are made twice a year, the yield to maturity is 11 percent.

Related Function: BONDPRICE.

BOOKOF

Syntax: BOOKOF(range)

Returns: String

Definition: Use BOOKOF to return the name of the worksheet referred to by range. If range does not refer to a specific worksheet, BOOKOF returns null.

Related Functions: COLOR, ROWOF, and ROWS.

BRIGHTNESS

Syntax: BRIGHTNESS(rgb_value)

Returns: 0 through 1

Definition: Use BRIGHTNESS to return the brightness component of a color expressed as an RGB value. In a method similar to the one used to specify the red, green, and blue values of a color (RGB), you can also specify the same color using hue, saturation, and brightness. This function translates a RGB value into the brightness component of the HSB color.

Related Function: HSBTORGB, HUE, RGB, and SATURATION.

CEILING

Syntax: CEILING(number, significance)

Returns: Number

Definition: This function returns a number that is rounded up, from zero, to the nearest multiple of significance. Number is the value you want to round. Significance is the multiple to which you want to round.

Example	Result
CEILING(2.5,1)	3
CEILING(1.5,0.1)	1.5
CEILING(0.234,0.01)	0.24
CEILING(-2.5,-2)	-4

CELLS

Syntax: CELLS()

Returns: Number

Definition: Use CELLS to return the number of non-empty cells in the current workbook. Non-empty cells include those containing data, as well as those that contain no data but have been formatted.

Related Functions: COUNT, DCOUNT, FORMULAS, SHEETCELLS, and SHEETFORMULAS.

CELLTEXT

Syntax: CELLTEXT(cell)

Returns: String

Definition: Use CELLTEXT to return a string representation of the displayed contents of the specified cell.

Example	Result
CELLTEXT(A1)	2.95

In the example, A1 contains the value 2.95.

Related Functions: CELL, FORMULATEXT, INDEX, and INDIRECT.

CFCMPXFVALUE

Syntax: CFCMPXFVALUE(cflow_period, amounts, no_periods, interests, timing, comp_period)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the future value based on cash flows per year, three correlating ranges of cash flows, repeats at this value, and interest rates during these periods, the payment timing (0 = end of period, 1 = beginning of period), and compounding periods per year.

Example

CFCMPXFVALUE(12, amt_arr, np_arr, int_arr, 1,1)

Result

289712.23

Amt_arr is a range that contains three values: 500, -1000, 750. Np_arr is a range that contains three values 120, 48, 72. Int_arr is a range that contains three values: .13, .13, .09.

CFCMPXPVALUE

Syntax: CFCMPXPVALUE(cflow_period, amounts, no_periods, interests, timing, comp_period)

Returns: Number

Definition: This function returns the net present value from a series of evenly spaced cash flows based on cash flows per year, three correlating ranges of cash flows, repeats at this value, and the period's interest rate, payment timing (0 = end of period, 1 = beginning of period), and compounding periods per year.

Example

CFCMPXPVALUE(4, amt_arr, np_arr, int_arr, 1,1)

Result

-2477456.49

Amt_arr is a range that contains three values: -1250000, 25000, 1525000. Np_arr is a range that contains three values: 1, 19, 1. Int_arr is a range that contains three values: 0.12, 0.12, 0.12.

CFESCFVALUE

Syntax: CFESCFVALUE(no_cflows, cflow_period, interest, amount, timing, speriod_opt, comp_period, growth)

Returns: Number

Definition: This function returns the future value of a series of regular deposits that increase at a fixed rate based on number of cash flows, cash flows a year, interest rate, initial cash flow amount, deposit timing (0 = end of period, 1 = beginning of period), short period option (0 = compound, 1 = simple), and cash flow growth rate.

Example

CFESCFVALUE(60, 12, 0.0875, 100000, 1, 1, 1, 0.05)

Result

8204155.24

CFESCPVALUE

Syntax: CFESCPVALUE(no_pays, pay_period, interest, amount, timing, speriod_opt, comp_period, growth)

Returns: Number

Definition: This function returns the present value of a series of regular deposits increasing at a fixed rate based on number of payments, payments rate, interest rate, payment amount, payment timing (0 = end of period, 1 = beginning of period), short period option (0 = compound, 1 = simple), compounding periods per year, and payment growth rate.

Example

CFESCPVALUE(240, 12, 0.0967, 1500, 1, 0, 12, 0.055)

Result

236569.92

Chapter 3: Alphabetical List of Functions

CFFESCINTEREST

Syntax: CFFESCINTEREST(fut_value, dep_period, amount, no_deposits, comp_period, speriod_opt, timing, growth)

Returns: Number

Definition: This function returns the interest rate needed to accumulate a given future value in an escalating series of deposits based on future value, deposits a year, deposit amount, number of deposits, compounding periods per year, short period option (0 = compound, 1 = simple), deposit timing (0 = end of period, 1 = beginning of period), and deposit growth rate.

Example

CFFESCINTEREST(80000, 12, 200, 180, 1, 0, 1, 0.05)

Result

6.23%

CFFINITDEPOSIT

Syntax: CFFESCINTEREST(no_pay, pay_period, interest, fut_value, timing, speriod_opt, comp_period, growth)

Returns: Number

Definition: This function returns the first payment in a series of increasing deposits to achieve a specified future value based on number of payments, payments per year, interest rate, future value, payment timing (0 = compound, 1 = simple), compounding periods per year, and deposit growth rate.

Example

CFFINITDEPOSIT(180, 12, 0.095, 65000, 1, 0, 365, 0.06)

Result

115.82

Cf_arr is a range that contains six values: 2600000, 2200000, 2300000, 2300000, 2800000, 2400000.

CFFPAYMENTS

Syntax: CFFPAYMENTS(fut_value, dep_period, amount, interest, comp_period, speriod_opt, growth)

Returns: Number

Definition: This function returns the required number of deposits in an escalating series based on future value, deposits a year, deposit amount, interest rate, compounding periods per year, deposit timing (0 = end of period, 1 = beginning of period), short period option (0 = compound, 1 = simple), and deposit growth rate.

Example	Result
CFFPAYMENTS(50000, 12, 200, 0.12, 1, 0, 1, 0.12)	102.00

CFFVALUE

Syntax: CFFVALUE(no_cflows, cflow_period, interest, cash_flows, timing, speriod_opt, comp_period)

Returns: Number

Definition: This function returns the future value from a series of evenly spaced cash flows and a fixed interest rate based on cash flows per year, interest rate, a range of cash flows, payment timing (0 = end of period, 1 = beginning of period), short period option (0 = compound, 1 = simple), and compounding periods per year.

Example	Result
CFFVALUE(26, 0.075, cf_arr, 1, 0, 365)	2197.14

Cf_arr is a range that contains six values: 250, 400, 475, 400, 400, 250.

Chapter 3: Alphabetical List of Functions

CFFXCINTEREST

Syntax: CFFXCINTEREST(cmpop_period, amounts, perlens, prates, seed, timing, fut_value, invtmnt)

Returns: Number

Definition: This function returns the interest rate required to accumulate a future value based on cash flow period per year, three correlating ranges of cash flow amounts, compounding periods until next cash flow, and period's interest rate, seed interest rate, cash flow timing (0 = end of period, 1 = beginning of period), target future value, and investment term (0 = duration of analysis, 1 = current period).

Example

CFFXCINTEREST(365, amt_arr, per_arr, int_arr, 0.055, 1, 78993.26, 1)

Result

5.57%

Amt_arr is a range or that contains four values: 20000, 10000, 15000, 30000. Per_arr is a range that contains four values: 78, 156, 189, 141. Int_arr is a range that contains four values: 0.06, 0.055, 0.05, 0.06.

CFPESCINTEREST

Syntax: CFPESCINTEREST(pres_value, cflow_period, amount, no_cflows, comp_period, speriod_opt, timing, growth)

Returns: Number

Definition: This function returns the interest rate required to reduce a cash flow changing by a fixed amount to a specified net present value based on present value, cash flows per year, initial cash flow amount, number of cash flows, compounding periods per year, short period option (0 = compound, 1 = simple), cash flow timing (0 = end of period, 1 = beginning of period), and cash flow growth rate.

Example

CFPESCINTEREST(100000, 12, 750, 240, 365, 0, 0, 0.055)

Result

11.55%

CFPINITDEPOSIT

Syntax: CFPINITDEPOSIT(no_pays, pay_period, interest, pres_value, timing, speriod_opt, comp_period, growth)

Returns: Number

Definition: This function returns the first deposit in a series of increasing deposits to achieve a specified present value based on number of payments, payments per year, interest rate, present value, payment timing (0 = end of period, 1 = beginning of period), short period option (0 = compound, 1 = simple), compounding periods per year, and deposit growth rate.

Example	Result
CFPINITDEPOSIT(180, 12, 0.0975, 150000, 1, 0, 365, 0)	1579.66
CFPINITDEPOSIT(180, 12, 0.0975, 150000, 1, 0, 365, 0.05)	1198.19

CFPINTEREST

Syntax: CFPINTEREST(no_cflows, cflow_period, cash_flows, seed, timing, speriod_opt, comp_period, pres_value)

Returns: Number

Definition: This function returns the internal rate of return from a series of evenly spaced cash flows with a fixed discount rate. The calculation is based on number of cash flows, cash flows per year, a range of cash flows, seed interest rate, cash flow timing (0 = end of period, 1 = beginning of period), short period option (0 = compound, 1 = simple), compounding periods per year, and present value.

Chapter 3: Alphabetical List of Functions

Example

CFPINTEREST(4, cf_arr, 0.12, 1, 0, 1, 0)

Result

25.09%

Cf_arr is a range that contains 13 values: -400000, -400000, -275000, 150000, 150000, 150000, 150000, 150000, 150000, 150000, 150000, 150000, 200000.

CFPPAYMENTS

Syntax: CFPPAYMENTS(press_value, pay_period, amount, interest, comp_period, timing, speriod_opt, growth)

Returns: Number

Definition: This function returns the number of payments in an escalating series based on present value, payments per year, payment amount, interest rate, compounding periods per year, payment timing (0 = end of period, 1 = beginning of period), short period option (0 = compound, 1 = simple), and deposit growth rate.

Example

CFPPAYMENTS(150000, 12, 1372.11, 0.105, 12, 0, 0, 0.04)

Result

172.00

CFPVALUE

Syntax: CFPVALUE(interest, cflow_period, cash_flows, timing, speriod_opt, comp_period)

Returns: Number

Definition: This function returns the present value from evenly spaced cash flows based on a fixed interest rate, cash flows per year, a range of cash flows, payment timing (0 = end of period, 1 = beginning of period), short period option (0 = compound, 1 = simple), and compounding periods per year.

Chapter 3: Alphabetical List of Functions

Example

CFPVALUE(0.12, 4, cf_arr, 1, 0, 1)

Result

1074961.252

Cf_arr is a range that contains 13 values: -400000, -400000, -275000, 150000, 150000, 150000, 150000, 150000, 150000, 150000, 150000, 150000, 300000.

CFPXCTINTEREST

Syntax: CFPXCTINTEREST(comp_period, amounts, perlens, prates, seed, timing, pres_value)

Returns: Number

Definition: This function returns the internal rate of return from a series of cash flows based on compounding periods per year, three correlating ranges of cash flow amounts, compounding periods before next cash flow, and period's interest rate, seed interest rate, cash flow timing (0 = end of period, 1 = beginning of period), and present value.

Example

CFPXCTINTEREST(365, amt_arr, per_arr, int_arr, 0.04, 1, 0)

Result

4.26%

Amt_arr is a range that contains six values: 25184, 2000, -6700, 2000, -1300, -21264.54. Per_arr is a range that contains six values: 2, 3, 12, 11, 3, 0. Int_arr is a range that contains six values: 0, 0, 0, 0, 0, 0.

CFXCTFVALUE

Syntax: CFXCTVALUE(amounts, perlens, prates, timing, comp_period, invtmnt)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the future value from a series of investments at specified intervals based on three correlating ranges of cash flow amount, compounding periods until next cash flow, and period's interest rate, payment timing (0 = end of period, 1 = beginning of period), compounding periods per year, and investment term (0 = duration of analysis, 1 = current period).

Example	Result
CFXCTFVALUE(amt_arr, per_arr, int_arr, 1, 365, 0)	79099.50
CFXCTFVALUE(amt_arr, per_arr, int_arr, 1, 365, 1)	78993.26

Amt_arr is a range that contains four values: 20000, 10000, 15000, 30000.

Per_arr is a range that contains four values: .06, .055, .05, .06.

CFXCTPVALUE

Syntax: CFXCTPVALUE(comp_period, amounts, perlens, prates, timing)

Returns: Number

Definition: This function returns the present value from a series of cash flows on specific dates based on compounding periods per year, three correlating ranges of cash flow amount, compounding periods until next cash flow, and interest rates during the period, and payment timing (0 = end of period, 1 = beginning of period).

CHAR

Syntax: CHAR(number)

Returns: String

Definition: Use CHAR to return the character set string corresponding to the decimal value of number. Number represents the value for a text character in the current application character set.

Chapter 3: Alphabetical List of Functions

This function returns the corresponding character as specified in the current application character set, ISO Latin 1 (8859-1).

Example	Result
CHAR(65)	A
CHAR(65 + 32)	a

In the first example, the character set value of 65 returns the corresponding text character, A.

In the second example, a value of 32 is added to the value of the uppercase A to obtain the lowercase a. The value needed to change the case of a letter may vary depending on the language used.

Related Function: CODE.

CHIDIST

Syntax: CHIDIST(x, degrees_freedom)

Returns: Number

Definition: This function returns the one-tailed probability of the chi-squared distribution. X is the value at which you want to evaluate the distribution. Degrees_freedom is the number of degrees of freedom.

Example	Result
CHIDIST(18.307,10)	0.050001

CHIINV

Syntax: CHIINV(actual_range, expected_range)

Returns: Number

Definition: This function returns the test for independence. CHITEST returns the value from the chi-squared distribution for the statistic and

Chapter 3: Alphabetical List of Functions

the appropriate degrees of freedom. Actual_range is the range that contains observations to test against expected values. Expected_range is the range that contains the ratio of the product of row totals and column totals to the grand total.

	A	B	C
1	Actual		
2		Men	Women
3	Agree	58	35
4	Neutral	11	25
5	Disagree	10	23
6			
7	Expected		
8		Men	Women
9	Agree	45.35	47.65
10	Neutral	17.56	18.44
11	Disagree	16.09	16.91

Example

CHITEST(B3..C5,B9..C11)

Result

0.000308

CHOOSE

Syntax: CHOOSE(number, expressions)

Returns: Number or string

Definition: Use CHOOSE to return the specified item (number) from the specified list (expressions). The first item in the list of expressions is counted as 0. If the item number specified is less than 0 or greater than the number of expressions, an error is returned. Expressions can contain a maximum of 49 items.

Example

CHOOSE(2, "zero", "two", "four", "six", "eight")
CHOOSE(2 * 2, 10, A1, A2, A3, B1, 5, B3)

Result

four
323.87

Chapter 3: Alphabetical List of Functions

In the first example, the first argument (2) is the number of the item in the list to choose. Since the count of expressions starts at "zero," the value returned is "four." In the second example, the number evaluates to four, so the value in the expression B1 is returned.

Related Functions: HLOOKUP, INDEX, and VLOOKUP.

CMONTH

Syntax: CMONTH(month_number)

Returns: String

Definition: Use CMONTH to return the name of the month corresponding to the given number of the month (month_number). If a number greater than 12 is used, the number is divided by 12 and the remainder is used as the argument.

Example	Result
CMONTH(2)	February
CMONTH(7 + 12)	July

In the first example, the argument 2 returns the second month in the year, February. In the second example, $7 + 12$ equals 19, which is divided by 12 leaving a remainder of 7, equal to the seventh month in the year, July.

Related Functions: CWEEKDAY, MONTH, and MONTHNAME.

CODE

Syntax: CODE(string)

Returns: Number

Definition: Use CODE to return the decimal equivalent of the character set value of the first character in the given string. This function returns

Chapter 3: Alphabetical List of Functions

the corresponding character as specified in the current application character set, ISO Latin 1 (8859-1).

Example	Result
CODE("Whale")	87
CODE("whale")	119

The first example returns the decimal value for character W, which is 87. The second example returns the decimal value for W (87) plus 32, the value that is added to an uppercase letter to obtain the lowercase equivalent of the letter.

Related Function: CHAR.

COL

Syntax: COL()

Returns: Number

Definition: Use COL to return the column number of the current worksheet cell. If a worksheet range is selected, the column number of the current cell in the range is returned. COL always returns the column number, even if column letters are displayed.

Example	Result
COL()	12
COL()	32768

In the first example, the current cell is L2. The value returned is the column number, 12 (the numeric equivalent of column L). In the second example, the current cell is AVLH1. The column number equivalent to column AVLH (32768) is returned.

Related Functions: COLOF, COLS, MAKECELL, MAKERANGE, ROW, ROWOF, and ROWS.

COLLATE

Syntax: COLLATE(string1, string2)

Returns: -1, 0, or 1

Definition: Use COLLATE to compare string1 and string2, according to the current collation table. Example comparisons and results are shown in the table below.

Example Comparison	Returns
String1 sorted before string2	-1
Both strings are equal	0
String2 sorted before string1	1

A collation table is a sequence used for sorting and merging alphabetic characters. This is sometimes referred to as a dictionary sort. Unlike the ASCII code sequence, collation table comparisons treat accented and non-accented versions of the same character as equal. All punctuation symbols are treated as equal when using the standard collation table.

The results in the following examples are dependent on the current collation type and table.

Example	Result
COLLATE("Whale", "whale")	1
COLLATE("a", "B")	-1
COLLATE("E", "a")	-1

The first example demonstrates that the COLLATE function is not case-sensitive. String1 (a) is sorted before string2 (B) in the second example. The third example sorts the character ^a (equivalent to a non-accented "a") prior to e.

Related Functions: COLLATIONITABLE, COLLATIONTYPE, and EXACT.

COLLATIONTABLE

Syntax: COLLATIONTABLE(number)

Returns: Range

Definition: Use COLLATIONTABLE to determine the contents and structure of a collation table. COLLATIONTABLE(0) returns a range containing the definition of the standard collation table. COLLATIONTABLE(1) returns a range containing the definition of the alternate collation table.

A collation table is a sequence used for sorting and merging alphabetic characters. The standard collation table treats uppercase, lowercase, accented, and non-accented versions of the same character as equal.

Related Functions: COLLATE and COLLATIONTYPE.

COLLATIONTYPE

Syntax: COLLATIONTYPE()

Returns: -2 through 2

Definition: Use COLLATIONTYPE to return a number representing the type of collation table and the number of passes currently being used to compare the strings. The current application character set is ISO Latin 1 (8859-1). This is a superset of the ASCII table.

A collation table is a sequence used for sorting and merging alphabetic characters. The default collation table is the standard table that treats as equal uppercase, lowercase, accented, and non-accented versions of the same letter. You can specify the use of a different collation table with the command `USE COLLATION TYPE number`. You can also specify the current application character set table, which sorts each character according to its character set (ASCII) value, or create your own sort

Chapter 3: Alphabetical List of Functions

order by defining an alternate collation table with the SET COLLATION TABLE range command.

You can also control the number of passes that are used to compare the values to be sorted against the current collation table. A double pass provides a finer granularity sort than a single pass.

The following table explains the values returned by COLLATIONTYPE.

Table	Pass	Returns
Alternate Table	Single	-2
Standard Table	Single	-1
Current Application Character Set Table	Single	0
Standard Table	Double	1
Alternate Table	Double	2

Related Functions: COLLATE and COLLATIONTABLE.

COLOF

Syntax: COLOF(range)

Returns: Number

Definition: Use COLOF to return the number of the first column in the specified worksheet range. Range can be any valid cell or range.

Example	Result
COLOF(A1)	1
COLOF(E3..J3)	5

The first example returns 1, the number for column A. The second example returns 5, the number for column E.

Related Functions: BOOKOF, COL, COLS, ROW, ROWOF, and ROWS.

Chapter 3: Alphabetical List of Functions

COLS

Syntax: COLS(range)

Returns: Number

Definition: Use COLS to return the number of columns in the specified worksheet range. Range can be any valid cell or range.

Example	Result
COLS(A1)	1
COLS(A1..D1)	4

The first example returns 1, the total number of columns in the range A1. The second example returns 4, the total number of columns in the range A1..D1.

Related Functions: COL, COLOF, ROW, ROWOF, and ROWS.

COMBIN

Syntax: COMBIN(number, number_chosen)

Returns: Number

Definition: This function returns the number of combinations for a given number of items. Use COMBIN to determine the total possible number of groups for a given number of items. Number is the number of items. Number_chosen is the number of items in each combination.

Example	Result
COMBIN(8,2)	28

COMPLEX

Syntax: COMPLEX(real_num, l_num, suffix)

Returns: Complex number as a string

Definition: This function converts real and imaginary coefficients into a complex number of the form $x + yi$ or $x + yj$. Real_num is the real coefficient of the complex number. l_num is the imaginary coefficient of the complex number. Suffix is the suffix for the imaginary component of the complex number. “i” and “j” are the only values allowed.

Example	Result
COMPLEX(3,4)	3 + 4i
COMPLEX(3,4,"j")	3 + 4j
COMPLEX(0,1)	i
COMPLEX(1,0)	1

CONFIDENCE

Syntax: CONFIDENCE(alpha, standard_dev, size)

Returns: Number

Definition: This function returns the confidence interval for a population mean. The confidence interval is a range on either side of a sample mean. Alpha is the significance level used to compute the confidence level. Standard_dev is the population standard deviation for the data range and is assumed to be known. Size is the sample size.

Example	Result
CONFIDENCE(0.05,2.5,50)	0.692951

CONTAINS

Syntax: CONTAINS(string, search_string)

Returns: 0 or 1

Definition: Use CONTAINS to determine if one string is found within another string. This function is case-sensitive. It returns 1 if search_string is a substring of string, or 0 if it is not. Both string and search_string can be variables or strings. Strings must be enclosed in a set of double quotation marks.

Example	Result
CONTAINS("February", "Feb")	1
CONTAINS("Bill Smith", "smith")	0

In the first example, "Feb" is a substring of "February" so a value of 1 is returned. In the second example, the search_string contains a lowercase "s" that is not found in string, so the function returns 0.

Related Functions: FIND, MATCH, and REPLACE.

CONVERT

Syntax: CONVERT(number, from_unit, to_unit)

Returns: Number

Definition: This function converts a number from one measurement system to another. For example, CONVERT can translate a table of distance in miles to a table of distances in kilometers. Number is the value in from_unit to convert. From_unit is the units for number. To_unit is the units for the result.

CONVERT accepts the following text values for from_unit and to_unit.

Chapter 3: Alphabetical List of Functions

Weight and mass	From_unit or to_unit
Gram	"g"
Slug	"sg"
Pound mass (avoirdupois)	"lbm"
U (atomic mass unit)	"u"
Ounce mass (avoirdupois)	"ozm"

Distance	From_unit or to_unit
Meter	"m"
Statute mile	"mi"
Nautical mile	"Nmi"
Inch	"in"
Foot	"ft"
Yard	"yd"
Angstrom	"ang"
Pica (1/72 in.)	"Pica"

Time	From_unit or to_unit
Year	"yr"
Day	"day"
Hour	"hr"
Minute	"mn"
Second	"sec"

Pressure	From_unit or to_unit
Pascal	"Pa"
Atmosphere	"atm"
mm of Mercury	"mmHg"

Force	From_unit or to_unit
Newton	"N"
Dyne	"dye"
Pound force	"lbf"

Chapter 3: Alphabetical List of Functions

Energy	From_unit or to_unit
Joule	"J"
Erg	"e"
Thermodynamic calorie	"c"
IT calorie	"cal"
Electron volt	"eV"
Horsepower-hour	"HPH"
Watt-hour	"Wh"
Foot-pound	"flb"
BTU	"BTU"

Power	From unit or to_unit
Horsepower	"HP"
Watt	"W"

Magnetism	From unit or to_unit
Tesla	"T"
Gauss	"ga"

Temperature	From_unit or to_unit
Degree Celsius	"C"
Degree Fahrenheit	"F"
Degree Kelvin	"K"

Liquid measure	From_unit or to_unit
Teaspoon	"tsp"
Tablespoon	"tbs"
Fluid ounce	"oz"
Cup	"cup"
Pint	"pt"
Quart	"qt"
Gallon	"gal"
Liter	"l"

The following abbreviated unit prefixes can be pre-pended to any metric from_unit or to_unit.

Chapter 3: Alphabetical List of Functions

Prefix	Multiplier	Abbreviation
exe	1E+18	"E"
peta	1E+15	"P"
tera	1E+12	"T"
giga	1E+09	"G"
mega	1E+06	"M"
kilo	1E+03	"k"
hecto	1E+02	"h"
deka	1E+01	"e"
deci	1E-01	"d"
centi	1E-02	"c"
milli	1E-03	"m"
micro	1E-06	"u"
nano	1E-09	"n"
pico	1E-12	"p"
femto	1E-15	"f"
atto	1E-18	"a"

Example	Result
CONVERT(1.0, "lbm", "kg")	0.453592
CONVERT(68, "F", "C")	20

CORREL

Syntax: CORREL(range1, range2)

Returns: Number

Definition: This function returns the correlation coefficient of the range1 and range2 cell ranges. Use the correlation coefficient to determine the relationship between two properties. Range1 and range2 are cell ranges of values.

	A	B	C	D	E
1	3	2	4	5	6
2	9	7	12	15	17

Example	Result
CORREL(A1..E1, A2..E2)	0.997054

Chapter 3: Alphabetical List of Functions

COS

Syntax: COS(number)

Returns: -1 through 1

Definition: Use COS to return the cosine of the specified number. Number can be any number given in radians. COS is the inverse of ACOS, which returns the arc cosine of a number. The following example returns the cosine of 1.243.

Example	Result
COS(1.243)	0.32

Related Functions: ACOS, ACOSH, ASIN, ASINH, ATAN, ATAN2, ATANH, COSH, SIN, SINH, TAN, and TANH.

COSH

Syntax: COSH(number)

Returns: Number greater than or equal to 1

Definition: Use COSH to return the hyperbolic cosine of the specified number.

Example	Result
COSH(.24)	1.03
COSH(10 - A5)	74.21

The first example returns the hyperbolic cosine of 0.24. In the second example, A5 contains the value 5, which is subtracted from 10. The result (74.21) is the hyperbolic cosine of 5.

Related Functions: ACOS, ACOSH, ASIN, ASINH, ATAN, ATAN2, ATANH, COS, SIN, SINH, TAN, and TANH.

COT

Syntax: COT(number)

Returns: Number

Definition: This function returns the cotangent of the given angle. Number is the angle in radians for which you want the cotangent.

Example	Result
COT(12)	-1.57

COUNT

Syntax: COUNT(expressions)

Returns: Number

Definition: Use COUNT to count the number of items (for example, strings, numbers, or worksheet cells in ranges) in expressions. Blank worksheet cells are not counted. Up to 50 arguments can be counted.

Example	Result
COUNT(1, "abc", 3, 4)	4
COUNT(A1, A2, A3..B3, B6, C9)	6

In the first example there are four arguments, so COUNT returns 4. All the cells specified in the second example are non-blank cells, so 6 worksheet cells are counted.

Related Functions: AVG, CELLS, and DCOUNT.

COUNTIF

Syntax: COUNTIF(range, criteria)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function counts the number of cells within a range that meet the given criteria. Range is the range of cells from which you want to count cells. Criteria is the criteria in the form of a number, expression, or text that defines which cells will be counted. For example, A3..A6 contains apples, oranges, peaches, and apples, respectively. The result is shown in the following example.

Example	Result
COUNTIF(A3..A6,"apples")	2

COUPDAYBS

Syntax: COUPDAYBS(settlement, maturity, frequency, basis)

Returns: Number

Definition: This function returns the number of days from the beginning of the coupon period to the settlement date. Settlement is the settlement date (the date after the issue date when the security is traded to the buyer). Maturity is the maturity date (the date when the security expires). Frequency is the number of payments per year. For annual payments, frequency =1; for semiannual, frequency 2; for quarterly, frequency =4. Basis is the type of day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example	Result
COUPDAYBS("1/25/93", "11/15/ 94",2,1)	71

COUPDAYS

Syntax: COUPDAYS(settlement, maturity, frequency, basis)

Returns: Number

Definition: This function returns the number of days in the coupon period that contains the settlement date. Settlement is the settlement date. The security settlement date is the date after the issue date when the security is traded to the buyer. Maturity is the maturity date. The maturity date is the date when the security expires. Frequency is the number of payments per year. For annual payments, frequency = 1; for semiannual, frequency = 2; for quarterly, frequency = 3; frequency = 4. Basis is the day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

COUPDAYS("1/25/93", "11/15/94", 2, 1)

Result

181

COUPDAYSNC

Syntax: COUPDAYSNC(settlement, maturity, frequency, basis)

Returns: Number

Definition: This function returns the number of days from the settlement date to the next coupon date. Settlement is the settlement date. The security settlement date is the date after the issue date when the security is traded to the buyer. Maturity is the maturity date. The maturity date is when the security expires. Frequency is the number of payments per year.

Chapter 3: Alphabetical List of Functions

For annual payments, frequency = 1; for semiannual, frequency =2; for quarterly, frequency =4. Basis is the type of day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

COUPDAYSN("1/25/93", "11/15/94",2,1)

Result

110

COUPNCD

Syntax: COUPNCD(settlement, maturity, frequency, basis)

Returns: Number

Definition: This function returns a number that represents the next coupon date after the settlement date. Settlement is the settlement date. The security settlement date is the date after the issue date when the security is traded to the buyer. Maturity is the security's maturity date. The maturity date is the date when the security expires. Frequency is the number of payments per year. For annual payments, frequency = 1; for semi-annual, frequency = 2; for quarterly, frequency = 4. Basis is the day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

COUPNCD("1/25/93", "11/15/94",2,1)

Result

34104.00

COUPNUM

Syntax: COUPNUM(settlement, maturity, frequency, basis)

Returns: Number

Definition: This function returns the number of coupons payable between the settlement date and maturity date, rounded up. Settlement is the settlement date. The security settlement date is the date after the issue date when the security is traded to the buyer. Maturity is the security's maturity date. The maturity date is the date when the security expires. Frequency is the number of payments per year. For annual payments, frequency = 1; for semiannual, frequency = 2; for quarterly, frequency = 4. Basis is the type of day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/160
3	Actual/365
4	European

Example

COUPNUM("1/25/93","11/15/94",2,1)

Result

4

COUPPCD

Syntax: COUPPCD(settlement, maturity, frequency, basis)

Returns: Number

Definition: This function returns a number that represents the previous coupon date before the settlement date. Settlement is the settlement date. The security settlement date is the date after the issue date when the security is traded to the buyer. Maturity is the security's maturity date. The maturity date is the date when the security expires. Frequency is the number of payments per year. For annual payments, frequency = 1; for

Chapter 3: Alphabetical List of Functions

semiannual, frequency = 2; for quarterly, frequency = 4. Basis is the type of day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

COUPPCD ("1/25/93", "11/15/94",2,1)

Result

33923 or November 15, 1992

COVAR

Syntax: COVAR(range1, range2)

Returns: Number

Definition: This function returns covariance, the average of the products of deviations for each data point pair. Use covariance to determine the relationship between two data sets. Range1 and range2 are ranges of integers. In the following example, range1 is A(x) = cells 3,2,4,5,6 and range2 is B(x) = cells (9,7,12,15,17).

Example

COVAR(A,B)

Result

5.2

CRITBINOM

Syntax: CRITBINOM(trials, probability_s, alpha)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the smallest value for which the cumulative binomial distribution is greater than or equal to a criterion value. Trials is the number of Bernoulli trials. Probability_s is the probability of a success. Alpha is the criterion value.

CTERM

Syntax: CTERM(interest, future_value, present_value)

Returns: Number

Definition: Use CTERM to calculate the number of compounding periods required for present_value to grow to the given future_value, based on a periodic interest rate.

Example	Result
CTERM(.12 / 12, 25000, 5000)	161.75

In the example, an investment currently worth \$5,000 and earning 12% annual interest compounded monthly (.12/12) will grow to \$25,000 in 161.75 months.

Related Functions: FV, FVL, LOANTERM, NPV, PV, PVL, RATE, and TERM.

CUMIPMT

Syntax: CUMIPMT(rate, nper, pv, start, end, type)

Returns: Number

Definition: This function returns the cumulative interest paid on a loan between start and end periods. Rate is the interest rate. Nper is the total number of periods. Pv is the present value. Start is the first period. End is the last period in the calculation. Type is the timing of the payment.

Chapter 3: Alphabetical List of Functions

Type	Timing
0	Payment at the end of the period.
1	Payment at the beginning of the period.

Example	Result
CUMIPMT(0.0075,360,125000, 1,1,0)	-937.50

CUMPRINC

Syntax: CUMPRINC(rate, nper, pv, start, end, type)

Returns: Number

Definition: This function returns the cumulative principal paid on a loan between start and end periods. Rate is the interest rate. Nper is the total number of periods. Pv is the present value. Start is the first period. End is the last period. Type is the timing of the payment.

Type	Timing
0	Payment at the end of the period
1	Payment at the beginning of the period

Example	Result
CUMPRINC(0.0075,360,125000,1,1 ,0)	-68.27827

CURRENCY

Syntax: CURRENCY(number)

Returns: String

Definition: Use CURRENCY to convert number into a string that includes a currency symbol, value, and two decimal places. You can use CURRENCY to format function results displayed in worksheets, notes, legends, and charts.

Example	Result
CURRENCY(41.446)	\$41.45
CURRENCY(C12 / 4)	\$9,500.00

In the first example, the number is rounded to two decimal places and given a currency sign. In the second example, cell C12 (38000) is divided by 4. The result is then displayed in the currency format as a string.

CUSTOMFORMAT

Syntax: CUSTOMFORMAT()

Returns: String

Definition: Use CUSTOMFORMAT to return the custom format string of the current worksheet cell. If there is no custom format string, null is returned. If more than one cell is selected, this function returns the custom format string of the first cell in the selected range (or null if the first cell does not contain a custom format).

Related Functions: CUSTOMFORMAT, FORMAT, NFORMAT, and VALIDCUSTOMFORMAT.

CUSTOMFORMATS

Syntax: CUSTOMFORMATS()

Returns: Range

Definition: Use CUSTOMFORMATS to return a range of the custom formats defined for the current worksheet.

Related Functions: CUSTOMFORMAT, FORMAT, NFORMAT, and VALIDCUSTOMFORMAT.

Chapter 3: Alphabetical List of Functions

CWEEKDAY

Syntax: CWEEKDAY(day_number)

Returns: String

Definition: Use CWEEKDAY to return the weekday name that corresponds to the number specified by day_number. Day_number can be any number from 1 (Sunday) through 7 (Saturday).

Example	Result
CWEEKDAY(4)	Wednesday
CWEEKDAY(A8 + 5)	Friday
CWEEKDAY(8)	Sunday

The first example returns the fourth day of the week, Wednesday. In the second example, A8 contains the value 1 which is added to 5 to return the sixth day of the week.

The last example shows that if day_number is greater than 7 (the total number of days in a week), it is divided by 7 and the weekday is determined by the remainder of the division.

Related Functions: CMONTH, DAY, DAYNAME, MONTH, and MONTHNAME.

CYAN

Syntax: CYAN()

Returns: 16776960

Definition: Use CYAN to return the RGB value for pure cyan.

Related Functions: BLACK, BLUE, GREEN, GREY, MAGENTA, RED, RGB, WHITE, and YELLOW.

DATE

Syntax: DATE(year, month, day)

Returns: Serial date number

Definition: Use DATE to return a serial date number based on the **year**, **month**, and **day** arguments provided. The three arguments can be any number within their respectively defined ranges. A serial date number is a number (from 0 to 401768) that indicates the number of days since December 30, 1899.

Example	Result
DATE(1900, 1, 1)	2.0
DATE(1957, 3, 12)	20891

In the first example, January 1, 1900 returns the serial date number 2.0. In the second example, March 12, 1957 returns the serial date number 20891.

Related Functions: ADATE, ATIME, DATEVALUE, NOW, TIME, TIMEVALUE, and YEAR.

DATEVALUE

Syntax: DATEVALUE(date_string)

Returns: Serial date number

Definition: Use DATEVALUE to return a serial date number that corresponds to the specified date_string. Date_string must be surrounded by a set of double quotation marks. A serial date number is a number (from 0 to 401768) that indicates the number of days since December 30, 1899.

Chapter 3: Alphabetical List of Functions

Example	Result
DATEVALUE("24 Aug 57")	21056
DATEVALUE("09/11/84")	30936

The first example returns 21056, the serial date number for August 24, 1957. In the second example, the serial date number for September 11, 1984 is returned, which is the value 30936.

Related Functions: ADATE, ATIME, DATE, MONTHNAME, NOW, and YEAR.

DAVG

Syntax: DAVG(database, offset, criteria)

Returns: Number

Definition: Use DAVG to return the average of the values in the offset column of the database range that match the requirements specified in the criteria range. The offset for the first column of the database range is 0.

The following example uses the database and criteria shown in the illustration to determine the average pay rate for department 12.

	A	B	C	D	E
1	Employee	Dept.	Hours	Pay Rate	Total Pay
2	M. Boyer	22.00	40.00	5.00	200.00
3	M. Ellis	12.00	38.00	5.30	201.40
4	N. Johnston	82.00	40.00	6.20	248.00
5	D. Phillips	12.00	40.00	8.00	320.00
6	C. Trent	22.00	40.00	5.25	210.00
7	J. Vargas	82.00	40.00	6.20	248.00
8	C. Wilson	12.00	40.00	5.30	212.00
9					
10	Employee	Dept.	Hours	Pay Rate	Total Pay
11		12			
12					
13				6.20	

Chapter 3: Alphabetical List of Functions

Example	Result
DAVG(A1..E8, 3, A10..E11)	6.20

In the example, the database range is A1..E8. The offset is 3, the Pay Rate column. The criteria range is A10..E11, which contains the value 12 in the Dept. column. DAVG calculates the average Pay Rate (offset) for the employees working in department 12.

Related Functions: AVG, DCOUNT, DSUM, STDEV, and SUM.

DAY

Syntax: DAY(date_number)

Returns: 1 through 31

Definition: Use DAY to return the day of the month for the specified serial date number (date_number). A serial date number is a number (from 0 to 401768) that indicates the number of days since December 30, 1899.

Example	Result
DAY(21056)	24
DAY(38053)	7
DAY(DATEVALUE("09/11/84"))	11

The first example returns 24, the day in the serial date number 21056. The second example returns 7, the day in the serial date number 38053. The third example uses the DATEVALUE function to return the serial date number (30936) of September 11, 1984. DAY returns 11, the day in the serial date number.

Related Functions: ADDDAYS, CWEEKDAY, DAYNAME, MONTH, and YEAR.

Chapter 3: Alphabetical List of Functions

DAYNAME

Syntax: DAYNAME(date_number)

Returns: String

Definition: Use DAYNAME to return the name of the weekday for the specified serial date number (date_number). A serial date number is a number (from 0 to 401768) that indicates the number of days since December 30, 1899.

Example	Result
DAYNAME(21056)	Saturday
DAYNAME(38055)	Tuesday

The first example returns Saturday, the name of the weekday in the serial date number 21056. The second example returns Tuesday.

Related Functions: ADDDDAYS, CWEEKDAY, and DAY.

DB

Syntax: DB(cost, salvage, life, period, month)

Returns: Number

Definition: This function returns depreciation for a specified period using the fixed-declining balance method. Cost is the initial cost of the invest. Salvage is the salvage value. Life is the useful life. Period is the period of depreciation. Month is the number of months in the first year.

Example	Result
DB(1000000,100000,6,1,7)	\$186,083

Chapter 3: Alphabetical List of Functions

DBTDISCOUNT

Syntax: DBTDISCOUNT(cost, mat_days, fvalue, cal)

Returns: Number

Definition: This function returns the discount rate of a note based on cost, days to maturity, face value, and calendar type (0 = 365 days/year, 1 = 360 days/year, 30 day months).

Example	Result
DBTDISCOUNT(99.63125, 15, 100, 0)	8.97%
DBTDISCOUNT(99.63125, 15, 100, 1)	8.85%

DBTDVALUE

Syntax: DBTDVALUE(fvalue, mat_days, disc_rate, cal)

Returns: Number

Definition: This function returns the value of a note based on face value, days to maturity, discount rate, and calendar type (0 = 365 days/year, 1 = 360 days/year, 30 day months).

Example	Result
DBTDVALUE(1000, 182, 0.0779, 1)	960.62

DBTENDBAL

Syntax: DBTENDBAL(start_bal, start_date, end_date, comp_period, speriod_opt, interest)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the ending balance from an interest bearing short term deposit, using 30 day month and 360 days/ year, based on beginning balance, start date, end date, compounding periods per year, short period option (0 = compound, 1 = simple) and interest rate.

Example

DBTENDBAL(100000, date(90,6,15), date(91,12,15),
365, 0, .085)

Result

113811.53

DBTINTEREST

Syntax: DBTINTEREST(cost, mat_days, fvalue, cal)

Returns: Number

Definition: This function returns the annual rate of a note based on cost, days to maturity, face value, and calendar type (0=365 days/year, 1=360 days/year, 30 day months).

Example

DBTINTEREST(99.63125, 15, 100, 1)

Result

8.88%

DBTVALUE

Syntax: DBTVALUE(cost, mat_days, fvalue, cal)

Returns: Number

Definition: This function returns the value of a discounted note based on days to maturity, expected return, and calendar type (0 = 365 days/year, 1 = 360 days/year, 30 day months).

Example

DBTVALUE(91, 0.07125, 0)

Result

98.25

Chapter 3: Alphabetical List of Functions

DCOUNT

Syntax: DCOUNT(database, offset, criteria)

Returns: Number

Definition: Use DCOUNT to return the number of non-blank cells in the offset column of the database range that meet the specified criteria range. The offset for the first column of the database is 0.

	A	B	C	D	E
1	Employee	Dept.	Hours	Pay Rate	Total Pay
2	M. Boyer	22.00	40.00	5.00	200.00
3	M. Ellis	12.00	38.00	5.30	201.40
4	N. Johnston	82.00	40.00	6.20	248.00
5	D. Phillips	12.00	40.00	8.00	320.00
6	C. Trent	22.00	40.00	5.25	210.00
7	J. Vargas	82.00	40.00	6.20	248.00
8	C. Wilson	12.00	40.00	5.30	212.00
9					
10	Employee	Dept.	Hours	Pay Rate	Total Pay
11		12		5.30	
12					
13				2.00	

Example

DCOUNT(A1..E8, 3, A10..E11)

Result

2.00

In the example, the database range is A1..E8. The criteria range is A10..E11, which contains 12 under Dept. and 5.30 in the Pay Rate column. DCOUNT counts the number of employees in department 12 who earn 5.30 an hour. The result is 2.00.

When all of the cells in the database are non-blank, the offset argument can be any column number in the database. If blank cells appear in the specified offset column, the row is not counted, even if it matches the criteria.

Related Functions: CELLS, COUNT, and DAVG.

Chapter 3: Alphabetical List of Functions

DDB

Syntax: DDB(cost, salvage, life, year_number)

Returns: Number

Definition: Use DDB to calculate the depreciation allowance for an asset, over a given period of time, using the double-declining balance method. Using this method, the depreciation rate is accelerated so the greatest depreciation occurs during the first period and declines during each subsequent period.

When the book value of the asset reaches the salvage value, the asset is fully depreciated. The book value for any given period of time is the total cost of the asset minus the depreciation charged over all prior periods. Cost and salvage can be any positive number. Life and year_number can be any positive integer.

A depreciation table that uses the DDB function to show the depreciation for the years of useful life is shown below. The DDB formula in cell B9 (formula test is shown in D9) contains absolute references to the cells that contain machine cost, salvage value, and useful life. The formula is copied down for all of the cells of the table. The examples below clarify these formulas. As shown in the examples, the amount of depreciation is calculated at \$2186.59 for the third year and \$789.02 for the sixth year. The DDB function is used to calculate a depreciation schedule for a machine with an original cost of \$15,000, a salvage value of \$2,000, and a useful life of seven years.

Chapter 3: Alphabetical List of Functions

	A	B	C	D
1				
2	Machine Cost	\$15000.00		
3	Salvage Value	\$2000.00		
4	Useful Life	7		
5				
6				
7		Amount of		
8	Year	Depreciation		DDB Formula
9	1	4285.71		=ddb(\$B\$2, \$B\$3, \$B\$4, A9)
10	2	3061.22		=ddb(\$B\$2, \$B\$3, \$B\$4, A10)
11	3	2186.59		=ddb(\$B\$2, \$B\$3, \$B\$4, A11)
12	4	1561.85		=ddb(\$B\$2, \$B\$3, \$B\$4, A12)
13	5	1115.61		=ddb(\$B\$2, \$B\$3, \$B\$4, A13)
14	6	789.02		=ddb(\$B\$2, \$B\$3, \$B\$4, A14)
15	7	0.00		=ddb(\$B\$2, \$B\$3, \$B\$4, A15)

Example

DDB(\$B\$2, \$B\$3, \$B\$4, 3)
DDB(\$B\$2, \$B\$3, \$B\$4, 6)

Result

2186.59
789.02

Related Functions: SLN and SYD.

DEC2BIN

Syntax: DEC2BIN(number, places)

Returns: Binary number as a string

Definition: This function converts a decimal number to binary. Number is the decimal integer you want to convert. Places is the number of characters to use. If number is omitted, the function uses the minimum number of necessary characters.

Example

DEC2BIN(9,4)
DEC2BIN(-100)

Result

1001
1110011100

Related Functions: BIN2DEC, BIN2HEX, BIN2OCT, DEC2HEX, DEC2OCT, HEX2BIN, HEX2DEC, HEX2OCT, OCT2BIN, OCT2DEC, and OCT2HEX.

Chapter 3: Alphabetical List of Functions

DEC2HEX

Syntax: DEC2HEX(number, places)

Returns: Hexadecimal number

Definition: This function converts a decimal number to hexadecimal. Number is the decimal integer you want to convert. Places is the number of characters to use. If places is omitted, the function uses the minimum number of necessary characters.

Example	Result
DEC2HEX(100,4)	0064
DEC2HEX(-54)	FFFFFFFFCA

Related Functions: BIN2DEC, BIN2HEX, BIN2OCT, DEC2BIN, DEC2OCT, HEX2BIN, HEX2DEC, HEX2OCT, OCT2BIN, OCT2DEC, and OCT2HEX.

DEC2OCT

Syntax: DEC2OCT(number, places)

Returns: Octal number

Definition: This function converts a decimal number to octal. Number is the decimal integer you want to convert. Places is the number of characters to use. If places is omitted, the function uses the minimum number of necessary characters.

Example	Result
DEC2OCT(58,3)	072
DEC2OCT(-100)	777777634

Related Functions: BIN2DEC, BIN2HEX, BIN2OCT, DEC2BIN, DEC2HEX, HEX2BIN, HEX2DEC, HEX2OCT, OCT2BIN, OCT2DEC, and OCT2HEX.

Chapter 3: Alphabetical List of Functions

DECTOFRAC

Syntax: DECTOFRAC(denominator, decimal, rounding)

Returns: Number

Definition: This function returns the decimal fraction of a value to the specified fractional base. The value is calculated based on the denominator, decimal number, and rounding method (0 = nearest, 1 = round up, 2 = round down).

Example	Result
DECTOFRAC(100, 1.375, 0)	38.00
DECTOFRAC(100, 1.375, 2)	37.00
DECTOFRAC(4, 2.675, 0)	3.00
DECTOFRAC(4, 2.675, 1)	3.00
DECTOFRAC(4, 2.675, 2)	2.00

DEGREES

Syntax: DEGREES(radians)

Returns: Number of degrees

Definition: Use DEGREES to convert the specified number of radians to degrees.

Example	Result
DEGREES(3.1415)	180
DEGREES(D5)	90

In the first example, 3.1415 radians are converted to 180 degrees. In the second example, cell D5 contains 1.57 which is converted to 90 degrees. In both examples, precision is set to 0.

Related Function: RADIANS.

Chapter 3: Alphabetical List of Functions

DELTA

Syntax: DELTA(number1, number2)

Returns: Number

Definition: This function determines whether two values are equal. Returns 1 if number1 = number2; otherwise, returns 0. Number1 is the first number. Number2 is the second number.

Example	Result
DELTA(5,4)	0
DELTA(5,5)	1
DELTA(0.5,0)	0

DEVSQ

Syntax: DEVSQ(number1, number2, ...)

Returns: Number

Definition: This function returns the sum of squares of deviations of data points from their sample mean. Number1, number2, ... are 1 to 50 arguments for which you want to calculate the sum of squared deviations.

Example	Result
DEVSQ(4,5,8,7,11,4,3)	48

DISC

Syntax: DISC(settlement, maturity, pr, redemption, basis)

Returns: Number

Chapter 3: Alphabetical List of Functions

	A	B	C	D	E
1	Employee	Dept.	Hours	Pay Rate	Total Pay
2	M. Boyer	22.00	40.00	5.00	200.00
3	M. Ellis	12.00	38.00	5.30	201.40
4	N. Johnston	82.00	40.00	6.20	248.00
5	D. Phillips	12.00	40.00	8.00	320.00
6	C. Trent	22.00	40.00	5.25	210.00
7	J. Vargas	82.00	40.00	6.20	248.00
8	C. Wilson	12.00	40.00	5.30	212.00
9					
10	Employee	Dept.	Hours	Pay Rate	Total Pay
11		12			
12					
13				8.00	

Example

DMAX(A1..E8, 3, A10..E11)

Result

8.00

In the example, the database range is A1..E8. The offset is 3, the Pay Rate column. The criteria range is A10..E11, which contains the value 12 in the Dept. column. DMAX calculates the maximum Pay Rate (offset) for employees working in department 12.

Related Functions: DMIN, MAX, and MIN.

DMIN

Syntax: DMIN(database, offset, criteria)

Returns: Number

Definition: Use DMIN to return the minimum value in the offset column of the database range that matches the requirements specified in the criteria range. The offset for the first column of the database range is 0. The following example uses the database and criteria worksheet ranges shown in the illustration to show the minimum pay for employees working 40 hours in department 12.

Chapter 3: Alphabetical List of Functions

	A	B	C	D	E
1	Employee	Dept.	Hours	Pay Rate	Total Pay
2	M. Boyer	22.00	40.00	5.00	200.00
3	M. Ellis	12.00	38.00	5.30	201.40
4	N. Johnston	82.00	40.00	6.20	248.00
5	D. Phillips	12.00	40.00	8.00	320.00
6	C. Trent	22.00	40.00	5.25	210.00
7	J. Vargas	82.00	40.00	6.20	248.00
8	C. Wilson	12.00	40.00	5.30	212.00
9					
10	Employee	Dept.	Hours	Pay Rate	Total Pay
11		12	40.00		
12					
13					212.00

Example **Result**
DMIN(A1..E8, 4, A10..E11) 212.00

In the example, the database range is A1..E8. The offset is 4, the Total Pay column. The criteria range is A10..E11, which contains the value 12 in the Dept. column and 40 in the Hours column. DMIN calculates the minimum Total Pay (offset) for employees working in department 12 who worked 40 hours.

Related Functions: DMAX, MAX, and MIN.

DOLLARDE

Syntax: DOLLARDE(fractional_dollar, fraction)

Returns: Number

Definition: This function converts a dollar price expressed as a fraction into a dollar price expressed as a decimal number. Fractional_dollar is a number expressed as a fraction. Fraction is the integer to use in the denominator of the fraction.

Example **Result**
DOLLARDE(1.02,16) 1.125

Chapter 3: Alphabetical List of Functions

DOLLARFR

Syntax: DOLLARFR(decimal_dollar, fraction)

Returns: Number

Definition: This function converts a dollar price expressed as a decimal number into a dollar price expressed as a fraction. Decimal_dollar is a decimal number. Fraction is the integer to use in the denominator of the fraction.

Example	Result
DOLLARFR(1.125,16)	1.02

DPDBDEPRECIATION

Syntax: DPDBDEPRECIATION(life, fy_conv, cost, conversion, month, decl_rate)

Returns: Range

Definition: This function returns the declining balance depreciation based on lifetime, first year convention (0 = month start, 1 = half year, 2 = half month, 3 = month end, 4 = year start, 5 = year end, 6 = half quarter), cost, conversion flag (T/F), month placed in service, and declining rate.

Example	Result
DPDBDEPRECIATION(15, 1, 1500000, 1, 3, 1.5)	Resultrange(cell1) = 750000.00 Resultrange(cell16) = 442867.50

DPFYDEPRECIATION

Syntax: DPFYDEPRECIATION(fy_conv, month)

Returns: Number

Definition: This function returns the proportion of first year depreciation owed based on first year convention (0 = month start, 1 = half year, 2 = half month, 3 = month end, 4 = year start, 5 = year end, 6 = half quarter), and month placed in service.

Example	Result
DPFYDEPRECIATION(0, 4)	0.75
DPFYDEPRECIATION(0, 8)	0.42
DPFYDEPRECIATION(1, 4)	0.50

DPSLDEPRECIATION

Syntax: DPSLDEPRECIATION(life, fy_conv, cost, month)

Returns: Range

Definition: This function returns a straight line depreciation based on lifetime, first year convention (0 - month start, 1 = half year, 2 = half month, 3 = month end, 4 = year start, 5 = year end, 6 = half quarter), cost, and month placed in service.

Example	Result
DPSLDEPRECIATION(31.5, 2, 1500000, 3)	Resultrange(cell1) = 37698 Resultrange(cell32) = 33730

Chapter 3: Alphabetical List of Functions

DPSMDEPRECIATION

Syntax: DPSMDEPRECIATION(life, fy_conv, cost, month)

Returns: Range

Definition: This function returns a sum of years digits depreciation based on lifetime, first year convention (0 = month start, 1 = half year, 2 = half month, 3 = month end, 4 = year start, 5 = year end, 6 = half quarter), cost, and month placed in service.

Example

DPSMDEPRECIATION(5, 0, 22500, 6)

Result

Resultrange(cell1) = 4375.00

Resultrange(cell6) = 625.00

DSTD

Syntax: DSTD(database, offset, criteria)

Returns: Number

Definition: Use DSTD to determine the population standard deviation for the offset column values of the database that match the requirements specified in the criteria range. The offset for the first column of the database range is 0.

The following example uses the database and criteria worksheet ranges shown in the illustration. The population standard deviation of the pay rates for employees working in departments 12 and 22 is 1.12.

Chapter 3: Alphabetical List of Functions

	A	B	C	D	E
1	Employee	Dept.	Hours	Pay Rate	Total Pay
2	M. Boyer	22.00	40.00	5.00	200.00
3	M. Ellis	12.00	38.00	5.30	201.40
4	N. Johnston	82.00	40.00	6.20	248.00
5	D. Phillips	12.00	40.00	8.00	320.00
6	C. Trent	22.00	40.00	5.25	210.00
7	J. Vargas	82.00	40.00	6.20	248.00
8	C. Wilson	12.00	40.00	5.30	212.00
9					
10	Employee	Dept.	Hours	Pay Rate	Total Pay
11		12			
12		22			
13				1.12	

Example

DSTD(A1..E8, 3, A10..E12)

Result

1.12

In the example, the database range is A1..E8. The offset is 3, the Pay Rate column. The criteria range is A10..E12, which contains the values 12 and 22 in the Dept. column. DSTD calculates the population standard deviation of the Pay Rate for employees in departments 12 and 22.

Related Functions: DSTDEV, STD, and STDEV.

DSTDEV

Syntax: DSTDEV(database, offset, criteria)

Returns: Number

Definition: Use DSTDEV to return the sample standard deviation for the offset column values of the database that matches the requirements specified in the criteria range. The offset for the first column of the database range is 0.

Chapter 3: Alphabetical List of Functions

The following example uses the database and criteria worksheet ranges shown in the illustration. The total pay for employees working in departments 12 and 22 is \$1143.40.

	A	B	C	D	E
1	Employee	Dept.	Hours	Pay Rate	Total Pay
2	M. Boyer	22.00	40.00	5.00	200.00
3	M. Ellis	12.00	38.00	5.30	201.40
4	N. Johnston	82.00	40.00	6.20	248.00
5	D. Phillips	12.00	40.00	8.00	320.00
6	C. Trent	22.00	40.00	5.25	210.00
7	J. Vargas	82.00	40.00	6.20	248.00
8	C. Wilson	12.00	40.00	5.30	212.00
9					
10	Employee	Dept.	Hours	Pay Rate	Total Pay
11		12			
12		22			
13					1143.40

Example

DSUM(A1..E8, 4, A10..E12)

Result

\$1143.40

In the example, the database range is A1..E8. The offset is 4, the Total Pay column. The criteria range is A10..E12, which contains the values 12 and 22 in the Dept. column. DSUM totals the Total Pay column for employees in departments 12 and 22.

Related Functions: AVG, DAVG, and SUM.

DSUMSQ

Syntax: DSUMSQ(database, offset, criteria)

Returns: Number

Definition: Use DSUMSQ to return the sum of the squares of the offset column values for the database range that match the requirements

Chapter 3: Alphabetical List of Functions

specified in the criteria range. The offset for the first column of the database range is 0,

The following example uses the database and criteria worksheet ranges shown in the illustration. The sum of the squares for test scores in the morning class equals 39,962.28.

	A	B	C	D
1	Test Scores	Morning Class	Afternoon Class	Average
2	Student 1	83.01	96.80	89.91
3	Student 2	60.50	52.68	56.59
4	Student 3	40.67	75.40	58.04
5	Student 4	34.02	30.89	32.46
6	Student 5	45.23	66.89	56.06
7	Student 6	66.36	98.65	92.51
8	Student 7	43.29	97.79	70.54
9	Student 8	11.31	65.66	38.49
10	Student 9	99.81	52.96	75.89
11	Student 10	28.27	35.91	32.09
12	Student 11	62.06	33.65	47.86
13	Student 12	26.08	51.96	39.02
14				
15	Test Scores	Morning Class	Afternoon Class	Average
16		1.00		
17				
18		39,962.28		

Example	Result
DSUMSQ(A1..D13, 1, A15..D16)	40160.9011

In the example, the database range is A1..D13. The offset is 1, the Morning Class column. The criteria range is A15..D16, which contains the formula =TRUE() in the Morning Class column. DSUMSQ totals the square of each item in the Morning Class column.

Related Function: SUMSQ.

DTFPERIODS

Syntax: DTFPERIODS(start_date, end_date, periods)

Returns: Range

Definition: This function returns the number of financial periods between two dates based on start date, end date, and number of periods per year, using the 360 days per year convention.

Example

DTFPERIODS(DATEVALUE("03/29/90"),
DATEVALUE("01/14/95"), 2)

Result

9.58

DTPERIODS

Syntax: DTPERIODS(start_date, end_date, periods)

Returns: Range

Definition: This function returns the number of periods between two dates based on start date, end date, and number of periods per year.

Example

DTPERIODS(DATEVALUE("03/29/90"),
DATEVALUE("01/14/95"), 2)
DTPERIODS(DATEVALUE("09/12/92"),
DATEVALUE("12/25/92"), 365)

Result

9.59

104.00

DURATION

Syntax: DURATION(settlement, maturity, coupon_yld, frequency, basis)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the Macauley duration for an assumed par value of \$100. Duration is defined as the weighted average of the present value of the cash flows and is used as a measure of a bond price's response to changes in yield. Settlement is the settlement date. Maturity is the maturity date. Coupon is the coupon rate. Yld is the yield. Frequency is the number of payments. Basis is the day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

DURATION("1/1/86","1/1/94",0.08,0.09,2,1)

Result

5.993775

DVAR

Syntax: DVAR(database, offset, criteria)

Returns: Number

Definition: Use DVAR to return the population variance of the offset column values for the database range that meets the criterion specified in the criteria range. The offset for the first column of the database range is 0.

The following examples use the database and criteria worksheet ranges shown in the illustration. The numbers, 664.10 and 562.86, represent the population variance for each class.

Chapter 3: Alphabetical List of Functions

1	Test Scores	Morning Class	Afternoon Class	Average
2	Student 1	83.01	96.80	89.91
3	Student 2	60.50	52.68	56.59
4	Student 3	40.67	75.40	58.04
5	Student 4	34.02	30.89	32.46
6	Student 5	45.23	66.89	56.06
7	Student 6	86.36	98.65	92.51
8	Student 7	43.29	97.79	70.54
9	Student 8	11.31	65.66	38.49
10	Student 9	99.81	52.96	75.89
11	Student 10	28.27	35.91	32.09
12	Student 11	62.06	33.65	47.86
13	Student 12	26.08	51.96	39.02
14				
15	Test Scores	Morning Class	Afternoon Class	Average
16		1.00	1.00	
17				
18		664.10	562.86	

Example

DVAR(A1..D13, 1, A15..D16)

DVAR(A1..D13, 2, A15..D16)

Result

664.10

562.86

In both examples, the database range is A1..D13. The criteria range is A15..D16, which contains the formula =TRUE() in the Morning Class and Afternoon Class columns.

The first example uses an offset of 1 to return the population variance for the Morning Class column (664.10). An offset of 2 is used in the second example to return the population variance for the Afternoon Class column (562.86).

Related Functions: DVARIANCE, VAR, and VARIANCE.

DVARIANCE

Syntax: DVARIANCE(database, offset, criteria)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: Use DVARIANCE to return the sample variance of the offset column values for the database range that matches the requirements specified in the criteria range. The offset for the first column of the database range is 0.

The following examples use the database and criteria worksheet ranges shown in the illustration.

	A	B	C
1	Heart Rates	Before Experiment	After Experiment
2		71	84
3		67	72
4		71	70
5		78	85
6		65	71
7		70	81
8			
9			
10		Before Experiment	After Experiment
11		1.00	1.00
12			
13	Sample Variance	19.87	47.77

Example

DVARIANCE(B1..C7, 0, B10..C11)

DVARIANCE(B1..C7, 1, B10..C11)

Result

19.87

47.77

In both examples, the database range is B1..C7. The criteria range is B10..C11, which contains the formula =TRUE() in the Before Experiment and After Experiment columns.

The first example uses an offset of 0 to return the sample variance for the Before Experiment column (19.87). An offset of 1 is used in the second example to return the sample variance for the After Experiment column (47.77).

Related Functions: DVAR, VAR, and VARIANCE.

E

Syntax: E()

Returns: Number

Definition: Use E to return the value of the constant E (natural logarithmic base), which is approximately 2.718282.

Related Functions: EXP, LN, LOG, and LOGN.

EFFECT

Syntax: EFFECT(nominal_rate, npery)

Returns: Number

Definition: This function returns the effective annual interest rate, given the nominal annual interest rate and the number of compounding periods per year. Nominal_rate is the nominal interest rate. Npery is the number of compounding periods per year.

Example	Result
EFFECT(5.25%,4)	0.053543

ERF

Syntax: ERF(lower_limit, upper_limit)

Returns: Number

Definition: This function returns the error function integrated between lower_limit and upper_limit. Lower_limit is the lower bound; upper_limit is the upper bound.

Example	Result
ERF(0.74500)	0.70793

Chapter 3: Alphabetical List of Functions

ERFC

Syntax: ERFC(x)

Returns: Number

Definition: This function returns the complementary ERF function integrated between x and infinity. X is the lower bound for integrating ERF.

Example	Result
ERFC(1)	0.1573

ERR

Syntax: ERR()

Returns: ERR 26

Definition: Use ERR to return the error value, ERR 26. This function can be used as an argument in an IF function.

Example

IF(A1 = 1, "OK", ERR())

In the example, an IF function is placed in a worksheet cell and used to test whether the content of cell A1 is equal to 1. If the condition is true, OK is placed in the cell; otherwise, ERR 26 is placed in the cell.

Related Functions: ERRNR and ISERR.

ERRNR

Syntax: ERRNR(expression)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: Use ERRNR to return the number of the error generated by the expression. If no error is generated, 0 is returned.

Example	Result
ERRNR(DATEVALUE("12/32/90"))	18

In the example, DATEVALUE is given an invalid date format. ERR 18, which signifies a bad date, is returned by DATEVALUE, and ERRNR converts this error value to the number 18.

Related Functions: ERR and ISERR.

EVALUATE

Syntax: EVALUATE(string)

Returns: Number

Definition: Use EVALUATE to treat the specified string as a formula.

Example	Result
EVALUATE("32 + 51")	83

Related Functions: STR, STRING, and VALUE.

EVEN

Syntax: EVEN(number)

Returns: Number

Definition: This function returns a number that is rounded up to the nearest even integer. Number is the value to round.

Chapter 3: Alphabetical List of Functions

Example	Result
EVEN(1.5)	2
EVEN(3)	4
EVEN(2)	2
EVEN(-1)	-2

EXACT

Syntax: EXACT(string1, string2)

Returns: 0 or 1

Definition: Use EXACT to perform a case-sensitive comparison of two strings (string1 and string2) to determine whether they match. A value of 1 is returned if the two strings match exactly, 0 is returned if they do not.

Example	Result
EXACT("Bill Smith", "bill smith")	0
EXACT("Bill Smith", "Bill Smith")	1

In the first example, the second string does not use the same capitalization. Therefore, it does not match the first string and a value of 0 is returned. In the second example, the two strings match, so a 1 is returned.

Related Functions: COLLATE and FIND.

EXP

Syntax: EXP(number)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: Use EXP to calculate the value of E() raised to the power of the specified number. E() returns a constant value of approximately 2.718282.

Example	Result
EXP(1.1)	3.004
EXP(-1)	0.370

Related Functions: E and LN.

EXPONDIST

Syntax: EXPONDIST(x, lambda, cumulative)

Returns: Number

Definition: This function returns the exponential distribution. X is the value of the function. Lambda is the parameter value. Cumulative is a logical value that indicates which form of the exponential function to provide. If cumulative is TRUE, EXPONDIST returns the cumulative distribution function; if FALSE, it returns the probability density function.

Example	Result
EXPONDIST(0.2, 10, TRUE())	0.864665

EXPONENTIAL

Syntax: EXPONENTIAL(number)

Returns: Number

Definition: Use EXPONENTIAL to return a number selected randomly from an exponential distribution with a mean of number.

Chapter 3: Alphabetical List of Functions

Example

EXPONENTIAL(10)
EXPONENTIAL(-1)

Result

random number
random number

The first example returns a random number from a population of samples with a mean of 10. The second example returns a random number from a population of samples with a mean of -1 .

Related Functions: NORMAL, RAND, and UNIFORM.

EXPSMOOTH

Syntax: EXPSMOOTH(series, weights)

Returns: Number

Definition: This function transforms a series with exponential smoothing, based on a range that contains the series, and a range that contains weighted values.

Example

EXPSMOOTH(ser_arr, wght_arr)

Result

Result cell 1 = 12.6474
Result cell 11 = 12.6366

Ser_arr is a range that contains 20 values: 12, 12.75, 13, 12.5, 12.5, 12.75, 13.5, 13, 12.5, 12, 12.25, 12, 11.75, 12.25, 12.75, 12.5, 13, 13.5, 13.25, 12.25. Wght_arr is a range that contains 10 values: 1.00, 1.10, 1.21, 1.33, 1.46, 1.61, 1.77, 1.95, 2.14, 2.36.

FACTDOUBLE

Syntax: FACTDOUBLE(number)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the double factorial of a number. Number is the value for which to return the double factorial.

Example	Result
FACTDOUBLE(6)	48
FACTDOUBLE(7)	105

FACTORIAL

Syntax: FACTORIAL(number)

Returns: Number

Definition: Use FACTORIAL to convert number to an integer and return its factorial. ERR12 is returned if you use a negative argument with this function.

Example	Result
FACTORIAL(3)	6
FACTORIAL(0)	1

In the first example, the factorial of 3 is calculated as $3 * 2 * 1 = 6$. The second example returns 1 as the factorial of 0.

Related Function: SIGN.

FALSE

Syntax: FALSE()

Returns: 0

Definition: Use FALSE to return a value of 0. This function is used in conditional statements and functions. In the example below, A3 is not equal to 1, so the value of FALSE(0) is returned.

Chapter 3: Alphabetical List of Functions

Example	Result
IF(A3 = 1, TRUE(), FALSE())	0

Related Functions: IF and TRUE.

FDIST

Syntax: FDIST(x, degrees_freedom1, degrees_freedom2)

Returns: Number

Definition: This function returns the F probability distribution. X is the value at which to evaluate the function. Degrees_freedom1 is the numerator degrees of freedom. Degrees_freedom2 is the denominator degrees of freedom.

Example	Result
FDIST(15.20675,6,4)	0.01

FIND

Syntax: FIND(search_string, string, number)

Returns: Number

Definition: Use FIND to search for search_string within string, starting at the position specified by number. A starting position of 0 represents the first character of the string. FIND returns a number representing the position of search_string within string. ERR 12 is returned if the string is not found. Number can be any integer expression.

Example	Result
FIND("hi", "Ohio", 0)	1
FIND("Hi", "Ohio", 0)	ERR 12

Chapter 3: Alphabetical List of Functions

In the first example, "hi" is contained in the string "Ohio" starting at position 1, so 1 is returned. In the second example, "Hi" is not found, so ERR 12 is returned.

Related Functions: CONTAINS, EXACT, MATCH, and REPLACE.

FINV

Syntax: FINV(probability, degrees_freedom1, degrees_freedom2)

Returns: Number

Definition: This function returns the inverse of the F probability distribution. Probability is a probability associated with the F cumulative distribution. Degrees_freedom1 is the numerator degrees of freedom. Degrees_freedom2 is the denominator degrees of freedom.

Example	Result
FINV(0.01,6,4)	15.20686

FISHER

Syntax: FISHER(x)

Returns: Number

Definition: This function returns the Fisher transformation at x. X is a numeric value for which you want the transformation.

Example	Result
FISHER(0.75)	0.972955

Chapter 3: Alphabetical List of Functions

FISHERINV

Syntax: FISHERINV(y)

Returns: Number

Definition: This function returns the inverse of the Fisher transformation. Y is the value for which you want to perform the inverse of the transformation.

Example	Result
FISHERINV(0.972955)	0.75

FLOOR

Syntax: FLOOR(number, significance)

Returns: Number

Definition: This function rounds number down, toward zero, to the nearest multiple of significance. Number is the numeric value you want to round. Significance is the multiple to which you want to round.

Example	Result
FLOOR(2.5, 1)	2

FORECAST

Syntax: FORECAST(x, known_y's, known_x's)

Returns: Number

Definition: This function calculates, or predicts, a future value by using existing values. X is the data point for which you want to predict a value. Known_y's is the dependent range. Known_x's is the independent range.

Chapter 3: Alphabetical List of Functions

	A	B	C	D	E
1	6	7	9	15	21
2	20	28	31	38	40

Example

FORECAST(30,A1..E1,A2..E2)

Result

10.60725

FORMULAS

Syntax: FORMULAS()**Returns:** Number**Definition:** Use FORMULAS to return the number of cells that contain formulas in the current workbook.**Related Functions:** CELLS, SHEETCELLS, and SHEETFORMULAS.

FORMULATEXT

Syntax: FORMULATEXT(range)**Returns:** String**Definition:** Use FORMULATEXT to return the formula in the first cell of the specified worksheet **range** as a string. If the formula contains more than 255 characters, FORMULATEXT returns a range; each cell of the range contains 255 characters, except possibly the last. If the cell does not contain a formula, ERR 12 is returned.**Related Functions:** CELLTEXT and FORMULATEXTHIDDEN.

Chapter 3: Alphabetical List of Functions

FREQUENCY

Syntax: FREQUENCY(data_range, bins_range)

Returns: Range

Definition: This function calculates how often values occur within a range of values, and then returns a range of numbers. Data_range is any range of values for which you want to count frequencies. Bins_range is a range of intervals into which you want to group the value. The examples below use the following range of values.

	A	B
1	79	
2	85	
3	78	
4	85	70
5	83	79
6	81	89
7	95	
8	88	
9	87	

Example

FREQUENCY(A1..A9, B4..B6)

Result

Resultrange(cell1) = 0
Resultrange(cell2)] = 2
Resultrange(3) = 6
Resultrange(4)] = 1

FTEST

Syntax: FTEST(range1, range2)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the result of an F-test. An F-test returns the one-tailed probability that the variances in range1 and range2 are not significantly different. Range1 is the first range of data. Range2 is the second range of data.

	A	B	C	D	E
1	6	7	9	15	21
2	20	28	31	38	40

Example

FTEST(A1..E1, A2..E2)

Result

0.648318

FV

Syntax: FV(payment, interest, term)

Returns: Number

Definition: Use FV to calculate the future value of a series of payments (an annuity) based on a regular payment amount, a periodic interest rate, and a given term.

Example

FV(250, 0.09, 12)

FV(100, 0.105/12, 10*12)

Result

5035.18

21081.48

In the first example, if \$250 is deposited into an annuity every year for the next 12 years, at an annual interest rate of 9%, the annuity would have a value of \$5,035.18 at the end of 12 years. The second example shows that a yield of \$21,081.48 is obtained when \$100 per month is paid into an annuity for 10 years at an annual interest rate of 10.5%.

Related Functions: CTERM, FVL, NPV, PV, PVL, RATE, and TERM.

Chapter 3: Alphabetical List of Functions

FVDEPAMOUNT

Syntax: FVDEPAMOUNT(fut_value, dep_period, deposits, interest, comp_period, speriod_opt, timing)

Returns: Range

Definition: This function returns the size of deposits required to accumulate a specified future value based on future value, deposits per year, interest rates, compounding periods per year, short period option (0 = compound, 1 = simple), and deposit timing (0 = end of period, 1 = beginning of period).

Example	Result
FVDEPAMOUNT(50000, 24, 96, 0.075, 365, 0,0)	447.31

FVDEPOSITS

Syntax: FVDEPOSITS(fut_value, dep_period, amount, interest, comp_period, speriod_opt, timing)

Returns: Number

Definition: This function returns the number of deposits required to achieve a specified future value based on future value, deposits per year, deposit amount, interest rate, compounding periods per year, short period option (0 = compound, 1 = simple), and deposit timing (0 = end of period, 1 = beginning of period).

Example	Result
FVDEPOSITS(50000, 24, 300, 0.09, 12, 0, 1)	130.00

FVDRATE

Syntax: FVDRATE(fut_value, dep_period, amount, cash_flows, comp_period, speriod_opt, timing)

Returns: Range

Definition: This function returns the annual rate of return required to accumulate a specified amount based on future value, deposits per year, deposit amount, number of cash flows, compounding periods per year, short period option (0 = compound, 1 = simple), and deposit timing (0 = end of period, 1 = beginning of period).

Example

FVDRATE(500000, 12, 400, 240, 1, 0, 0)

Result

14.6%

FVGROWTHRATE

Syntax: FVGROWTHRATE(fut_value, no_periods, pres_value, comp_period)

Returns: Number

Definition: This function returns the annual interest rate required for change between present value, and future value based on future value, number of periods, present value, and compounding periods per year.

Example

FVGROWTHRATE(3000, 5475, 1000, 365)

Result

7.32%

FVGROWTHTIME

Syntax: FVGROWTHTIME(fut_value, interest, pres_value, speriod_opt, comp_period)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the number of periods required for a change between present and future value based on future value, interest rate, present value, short period option (0 = compound, 1 = simple), and compounding periods per year.

Example	Result
FVGROWTHTIME(3000, 0.075, 1000, 0, 365)	5347.13

FVINTEREST

Syntax: FVINTEREST(amount, interest, period, speriod_opt, comp_period)

Returns: Number

Definition: This function returns the interest earned on an amount left on deposit for a specified number of periods. The calculation is based on amount left, interest rate, years left, short period option (0 = compound, 1 = simple), and compounding periods per year.

Example	Result
FVINTEREST(10000, 0.079, 1, 0, 0)	822.04

FVL

Syntax: FVL(payment, interest, term)

Returns: Number

Definition: Use FVL to calculate the future value of a lump sum payment received today, based on the given term and periodic interest rate.

Example	Result
FVL(3000, 0.09, 12)	8437.99
FVL(100, 0.105, 20)	736.62

Chapter 3: Alphabetical List of Functions

The first example determines the yield of a \$3000 lump sum payment, earning an annual interest rate of 9% for 12 years. The second example determines the yield of \$100 earning an annual interest of 10.5% for 20 years.

Related Functions: CTERM, FV, NPV, PV, PVL, RATE, and TERM.

FVSCHEDULE

Syntax: FVSCHEDULE(principal, schedule)

Returns: Number

Definition: This function returns the future value of an initial principal after applying a series of compound interest rates. Principal is the present value. Schedule is a range of interest rates to apply.

	A	B	C
1	0.09	0.11	0.10

Example	Result
FVSCHEDULE(1,A1..C1)	1.33089

FVVALUE

Syntax: FVVALUE(dep_period, no_deposits, timing, amount, interest, speriod_opt, comp_period)

Returns: Number

Definition: This function returns the future value of a series of equal deposits occurring at fixed intervals based on deposits per year, number of deposits, deposit timing (0 = end of period, 1 = beginning of period),

Chapter 3: Alphabetical List of Functions

deposit amount, interest rate, short period option (0 = compound, 1 = simple), and compounding periods per year.

Example	Result
FVVALUE(12, 240, 0, 220, 0.135, 0, 1)	240287.01

GAMMADIST

Syntax: GAMMADIST(x, alpha, beta, cumulative)

Returns: Number

Definition: This function returns the gamma distribution. X is the value at which you want to evaluate the distribution. Alpha and beta are parameters to the distribution. Cumulative is a logical value that determines the form of the function. If cumulative is TRUE, GAMMADIST returns the cumulative distribution function; if FALSE, it returns the probability mass function.

Example	Result
GAMMADIST(10,9,2,FALSE())	0.032639

GAMMAINV

Syntax: GAMMAINV(probability, alpha, beta)

Returns: Number

Definition: This function returns the inverse of the gamma cumulative distribution. Probability is the probability associated with the gamma distribution. Alpha and beta are parameters to the distribution.

Example	Result
GAMMAINV(0.068094,9,2)	10

GAMMALN

Syntax: GAMMALN(x)

Returns: Number

Definition: This function returns the natural logarithm of the gamma function. X is the value for which you want to calculate GAMMALN.

Example	Result
GAMMALN(4)	1.791759

GCD

Syntax: GCD(number1, number2, ...)

Returns: Number

Definition: This function returns the greatest common divisor of two or more integers. Number1, number2, ... are 1 to 50 values.

Example	Result
GCD(5, 2)	1
GCD(24, 36)	12

GEOMEAN

Syntax: GEOMEAN(number1, number2, ...)

Returns: Number

Definition: This function returns the geometric mean of a range of positive data. Number1, number2, ... are 1 to 50 arguments for which you want to calculate the mean.

Example	Result
GEOMEAN(4,5,8,7,11,4,3)	5.476987

Chapter 3: Alphabetical List of Functions

GESTEP

Syntax: GESTEP(number, step)

Returns: Number

Definition: This function returns 1 if number is less than or equal to step; 0 is returned otherwise. Number is the value to test against step. Step is the threshold value.

Example	Result
GESTEP(5, 4)	1
GESTEP(-4, -5)	1
GESTEP(-1, 0)	0

GOAL

Syntax: GOAL(initial_estimate, result, formula)

Returns: Number

Definition: Use GOAL to attempt to find a value for an unknown term in an equation. This value makes the equation yield a desired result. You specify a formula with an unknown term (represented by the function GUESS()), a desired result, and an initial estimated value (initial_estimate for GUESS()).

Using your initial estimate, GOAL calculates the result of the formula and compares it with the desired result. Based on the difference between the actual and desired result, this function makes another estimate and calculates a new result. The process is iterated until either two successive results differ less than 1E- 7, or until the maximum of 20 recalculations is reached.

Chapter 3: Alphabetical List of Functions

Try to make your initial estimate closely approximate the value you expect for the variable GUESS(). Otherwise, the calculated result and the desired result may not converge sufficiently within 20 recalculations.

The initial_estimate and result can be any number, and the formula can be any numeric formula using GUESS() as a variable.

Example	Result
GOAL(1, 3, GUESS() ^ 2)	1.7321
GOAL(-1, 4, GUESS() ^ 2)	-2

In the first example, GOAL tries to determine what value, when raised to the power of two, returns a value of 3. The function begins with the estimate value of 1 and continues to calculate based on different guesses until it converges on a solution. The approximate result is 1.7321.

In the second example, the estimated value begins at -1 and, since the negative solution is closer to the estimate, the negative value is returned. In this example, both 2 and -2 are possible solutions to the problem.

Related Function: GUESS.

GREEN

Syntax: GREEN()

Returns: 65280

Definition: Use GREEN to return the RGB value for pure green.

Related Functions: BLACK, BLUE, CYAN, GREY, MAGENTA, RED, RGB, WHITE, and YELLOW.

Chapter 3: Alphabetical List of Functions

GREY

Syntax: GREY()

Returns: 8421504

Definition: Use GREY to return the RGB value for 50 percent grey.

Related Functions: BLACK, BLUE, CYAN, GREEN, MAGENTA, RED, RGB, WHITE, and YELLOW.

GROWTH

Syntax: GROWTH(known_y's, known_x's, new_x's, const)

Returns: Range

Definition: This function calculates predicted exponential growth by using existing data. Known_y's is the set of y-values you know in $y = b \cdot m^x$. Known_x's is an optional set of x-values. New_x's are new x-values. Const specifies whether to force the constant b to equal 1.

GUESS

Syntax: GUESS()

Returns: Number

Definition: Use GUESS as the independent variable in the formula supplied to the GOAL function. The result of the GUESS function is only defined when used with the GOAL function.

Related Function: GOAL

Chapter 3: Alphabetical List of Functions

HARMEAN

Syntax: HARMEAN(number1, number2, ...)

Returns: Number

Definition: This function returns the harmonic mean of a data set. Number1, number2, ... are 1 to 50 arguments for which you want to calculate the mean.

Example	Result
HARMEAN(4,5,8,7,11,4,3)	5.028376

HEX2BIN

Syntax: HEX2BIN(number, places)

Returns: Binary number as a string

Definition: This function converts a hexadecimal number to binary. Number is the hexadecimal number you want to convert. Places, which is optional, is the number of characters to use. If places is omitted, the function uses the minimum number of necessary characters.

Example	Result
HEX2BIN("F",8)	00001111
HEX2BIN("B7")	10110111

Related Functions: BIN2DEC, BIN2HEX, BIN2OCT, DEC2BIN, DEC2HEX, DEC2OCT, HEX2DEC, HEX2OCT, OCT2BIN, OCT2DEC, and OCT2HEX.

Chapter 3: Alphabetical List of Functions

HEX2DEC

Syntax: HEX2DEC(number)

Returns: Decimal number

Definition: This function converts a hexadecimal number to decimal. Number is the hexadecimal number you want to convert.

Example	Result
HEX2DEC("A5")	165
HEX2DEC("3DA408B9")	1034160313

Related Functions: BIN2DEC, BIN2HEX, BIN2OCT, DEC2BIN, DEC2HEX, DEC2OCT, HEX2BIN, HEX2OCT, OCT2BIN, OCT2DEC, and OCT2HEX.

HEX2OCT

Syntax: HEX2OCT(number, places)

Returns: Octal number as a string

Definition: This function converts a hexadecimal number to octal. Number is the hexadecimal number you want to convert. Places, which is optional, is the number of characters to use. If places is omitted, the function uses the minimum number of necessary characters.

Example	Result
HEX2OCT("F", 3)	017
HEX2OCT("3B4E")	35516

Related Functions: BIN2DEC, BIN2HEX, BIN2OCT, DEC2BIN, DEC2HEX, DEC2OCT, HEX2BIN, HEX2DEC, OCT2BIN, OCT2DEC, and OCT2HEX.

HLOOKUP

Syntax: HLOOKUP(search_item, range, offset)

Returns: Number or string

Definition: Use HLOOKUP to return the contents of a worksheet cell's position, which is determined by the given arguments.

HLOOKUP searches the top row of the specified worksheet range for the specified search_item. The lookup finds a cell in the top row that contains the closest value equal to or less than the search item. It then returns the contents of the cell that is offset the specified number of rows from the found cell. The row number of the found cell is offset 0. If you use an offset of 1, the returned value is the contents of the cell one row below the found cell.

An offset of 0 returns an index number (beginning with 0) of the located item if the search_item is a string. When a numeric search_item is supplied with an offset of 0, the cell contents is returned.

If the search_item is a string, the item must match the contents of a cell in the search row exactly. If no match is found, ERR 6 (Lookup Failed) is returned. When strings are used in the search row, no ordering is required.

Because the function stops the search as soon as it finds a match, numeric data in the search row must be in ascending order. The search_item and offset can be any number. If an offset exceeds the boundaries of the specified range, ERR 25 (Offset expression is outside the table range) is returned. The range can be any valid range. The following example uses the projected sales information, shown in the illustration, as the HLOOKUP table data.

Chapter 3: Alphabetical List of Functions

	A	B	C	D	E	F
1	SUBSIDIARY INCREASE	1993	1994	1995	1996	1997
2	South America - 8%	27,047	29,211	31,548	34,071	36,797
3	Europe - 30%	37,003	48,104	62,535	81,296	105,684
4	Asia - 30%	40,681	52,885	68,751	89,376	116,189
5	US - 20%	95,739	114,887	137,864	165,437	198,524
6						
7						
8			68,751			

Example

HLOOKUP(33000, B2..F5, 2)

Result

68,751

In this example, HLOOKUP determines the projected sales figure for Asia (using an offset of 2, beginning with 0) for the same time period when projected sales in South America are less than or equal to 33000. The closest matching figure in the South America row is in the 1995 column (31548). Using the offset of 2 (under column 1995), projected sales for Asia are 68,751.

Related Functions: CHOOSE, INDEX, and VLOOKUP.

HOUR

Syntax: HOUR(date_number)

Returns: Number

Definition: Use HOUR to calculate the hour component of a specified date number and return it in a 24-hour format. A serial date number is a number (from 0 to 401768) that indicates the number of days since December 30, 1899.

Example

HOUR(32000.4)

HOUR(TIMEVALUE("1:38P"))

Result

9

13

Chapter 3: Alphabetical List of Functions

The first example calculates the hour from the decimal portion (.4) of the date number, 32000.4. The result (9) represents 9:00 AM in a 24-hour format.

In the second example, 1:38P represents 1:38 PM. Since 1:38P is a string, TIMEVALUE is used to change it to a serial date number. HOUR evaluates the serial date number and returns the hour in a 24-hour format (13).

Related Functions: ADDHOURS, MINUTE, NOW, SECOND, TIME, and TIMEVALUE.

HSBTORGB

Syntax: HSBTORGB(hue, saturation, brightness)

Returns: RGB value

Definition: Use HSBTORGB to convert the components of a hue, saturation, and brightness specification to an RGB specification (such as, the red, green, and blue values of a color). Hue ranges from 0 to 360, while saturation and brightness values range from 0 to 1.

Related Functions: BRIGHTNESS, HUE, MIXEDCOLOR, RGB, and SATURATION.

HUE

Syntax: HUE(rgb_value)

Returns: Number in degrees

Definition: Use HUE to return the hue component of the rgb_value argument in relation to a color circle. HUE specifies the position on this color circle.

Chapter 3: Alphabetical List of Functions

In a method similar to the one used to specify the red, green, and blue values of a color (RGB), you can also specify the same color using hue, saturation, and brightness. This function translates a RGB value into the hue component of the HSB color.

The following table shows the positions of a number of sample colors.

Color	Position in degrees	Color	Position in degrees
Red	0	Cyan	180
Yellow	60	Blue	240
Green	120	Magenta	300

Related Functions: BRIGHTNESS, HSBTORGB, MIXEDCOLOR, RGB, and SATURATION.

HYPGEOMDIST

Syntax: HYPGEOMDIST(sample_s, number_sample, population_s, number_population)

Returns: Number

Definition: This function returns the hypergeometric distribution. HYPGEOMDIST returns the probability of a given number of sample successes, given the sample, size, population successes, and population size. Sample_s is the number of successes. Number_sample is the size of the sample. Population_s is the number of successes in the population. Number_population is the population size.

Example	Result
HYPGEOMDIST(1,4,8,20)	0.363261

Chapter 3: Alphabetical List of Functions

IF

Syntax: IF(condition, expression1, expression2)

Returns: Expression1 or expression2

Definition: Use IF to enter information in the worksheet cells that meet specific criteria. This function returns one of the two specified expressions, depending on whether the given condition is true or false. Condition is any logical expression. If condition is true (any non-zero number is considered to be true) expression1 is returned; otherwise, expression2 is returned.

Example	Result
IF(2 * 2 > 3, 100 * A4, 0)	700
IF(3 < 1, D1, MOD(E7, 3))	2

In the first example, A4 contains the value 7. In the second example, E7 contains the value 11 and D1 contains the value 2. The MOD function returns the remainder of the division ($E7 / 3$).

Related Functions: FALSE and TRUE.

IMABS

Syntax: IMABS(inumber)

Returns: Number

Definition: This function returns the absolute value (modulus) of a complex number in $x + yi$ or $x + yj$ text format. Inumber is a complex number for which you want the absolute value.

Example	Result
IMABS("5+12i")	13

Chapter 3: Alphabetical List of Functions

IMAGINARY

Syntax: IMAGINARY(inumber)

Returns: Number

Definition: This function returns the imaginary coefficient of a complex number in $x + yi$ or $x + yj$ text format. Inumber is a complex number for which you want the imaginary coefficient.

Example	Result
IMAGINARY("3+4i")	4

IMARGUMENT

Syntax: IMARGUMENT(inumber)

Returns: Number

Definition: This function returns the argument theta, an angle expressed in radians. Inumber is a complex number for which you want the argument theta.

Example	Result
IMARGUMENT("3+4i")	0.927295

IMCONJUGATE

Syntax: IMCONJUGATE(inumber)

Returns: String

Definition: This function returns the complex conjugate of a complex number in $x + yi$ or $x + yj$ text format. Inumber is a complex number for which you want the conjugate.

Example	Result
IMCONJUGATE("3+4i")	3 - 4i

IMCOS

Syntax: IMCOS(inumber)

Returns: Number

Definition: This function returns the cosine of a complex number in $x + yi$ or $x + yj$ text format. Inumber is a complex number for which you want the cosine.

Example

IMCOS("1+i")

Result

0.83373 - 0.988898i

IMDIV

Syntax: IMDIV(inumber1, inumber2)

Returns: String

Definition: This function returns the quotient of two complex numbers in $x + yi$ or $x + yj$ text format. Inumber1 is the complex numerator or dividend. Inumber2 is the complex denominator or divisor.

Example

IMDIV("-238+240i", "10+24i")

Result

5+12i

IMEXP

Syntax: IMEXP(inumber)

Returns: String

Definition: This function returns the exponential of a complex number in $x + yi$ or $x + yj$ text format. Inumber is a complex number for which you want the exponential.

Example

IMEXP("1+i")

Result

1.468694 + 2.287355i

Chapter 3: Alphabetical List of Functions

IMLN

Syntax: IMLN(inumber)

Returns: String

Definition: This function returns the natural logarithm of a complex number in $x + yi$ or $x + yj$ text format. Inumber is a complex number for which you want the natural logarithm.

Example

IMLN("3+4i")

Result

1.609438 + 0.927295i

IMLOG10

Syntax: IMLOG10(inumber)

Returns: String

Definition: This function returns the common logarithm (base 10) of a complex number in $x + yi$ or $x + yj$ text format. Inumber is a complex number for which you want the common logarithm.

Example

IMLOG10("3+4i")

Result

0.69897 +0.402719i

IMLOG2

Syntax: IMLOG2(inumber)

Returns: String

Definition: This function returns the base-2 logarithm of a complex number in $x + yi$ or $x + yj$ text format. Inumber is a complex number for which you want the base-2 logarithm.

Example

IMLOG2("3+4i")

Result

2.321928 + 1.337804i

IMPOWER

Syntax: IMPOWER(inumber, number)

Returns: String

Definition: This function returns a complex number raised to a power. Inumber is a complex number you want to raise to a power. Number is the power to which you want to raise the complex number.

Example	Result
IMPOWER("2+3i", 3)	-46 +9i

IMPRODUCT

Syntax: IMPRODUCT(inumber1, inumber2, ...)

Returns: String

Definition: This function returns the product of 2 or more complex numbers in $x + yi$ or $x + yj$ text format. Inumber1, inumber2, ... are 1 to 50 complex numbers to multiply.

Example	Result
IMPRODUCT("1+2i", 30)	30 +60i

IMREAL

Syntax: IMREAL(inumber)

Returns: Number

Definition: This function returns the real coefficient of a complex number in $x + yi$ or $x + yj$ text format. Inumber is a complex number for which you want the real coefficient.

Example	Result
IMREAL("6-9i")	6

Chapter 3: Alphabetical List of Functions

IMSIN

Syntax: IMSIN(inumber)

Returns: String

Definition: This function returns the sine of a complex number in $x + yi$ or $x + yj$ text format. Inumber is a complex number for which you want the sine.

Example

IMSIN("3+4i")

Result

3.853738 – 27.0168131i

IMSQRT

Syntax: IMSQRT(inumber)

Returns: String

Definition: This function returns the square root of a complex number in $x + yi$ or $x + yj$ text format. Inumber is a complex number for which you want the square root.

Example

IMSQRT("1+i")

Result

1.098684 + 0.45509i

IMSUB

Syntax: IMSUB(inumber1, inumber2, ...)

Returns: String

Definition: This function returns the difference of two complex numbers in $x + yi$ or $x + yj$ text format. Inumber1 is the complex number from which to subtract inumber2. Inumber2 is the complex number to subtract from inumber1.

Chapter 3: Alphabetical List of Functions

Example

IMSUB("13+4i", "5+3i")

Result

8 + i

IMSUM

Syntax: IMSUM(inumber1, inumber2, ...)

Returns: String

Definition: This function returns the sum of two or more complex numbers in $x + yi$ or $x + yj$ text format. Inumber1, inumber2, ... are 1 to 50 complex numbers to add.

Example

IMSUM("3+4i", "5-3i")

Result

8 + i

INDEX

Syntax: INDEX(range, column, row)

Returns: Number or string

Definition: Use INDEX to return the contents of a worksheet cell, where the cell's position is determined by the specified range, column offset, and row offset. Using the top left cell in the specified range as the reference point (column 0 and row 0), INDEX moves over the number of columns and down the number of rows you specify to return the contents of the specified cell.

The following illustration provides information for the examples below. The first cell of the range (A3) is column offset 0 and row offset 0.

Chapter 3: Alphabetical List of Functions

	A	B	C	D	E
1	Taxable		Married Filing	Married Filing	Head of
2	Income	Single	Jointly	Separately	Household
3	34000 - 34050	6,882	5,107	7,317	5,978
4	34050 - 34100	6,896	5,121	7,331	5,992
5	34100 - 34150	6,910	5,135	7,345	6,006
6	34150 - 34200	6,924	5,149	7,359	6,020
7	34200 - 34250	6,938	5,163	7,373	6,034
8	34250 - 34300	6,952	5,177	7,387	6,048
9	34300 - 34350	6,966	5,191	7,401	6,062
10	34350 - 34400	6,980	5,205	7,415	6,076

Example	Result
INDEX(A3..E10, 2, 4)	5163.00
INDEX(A3..E10, 1, 7)	6980.00

In the first example, column offset is 2 (Married Filing Jointly) and the row offset is 4. INDEX returns the contents of cell C7. The second example contains a column offset of 1 (Single) and a row offset of 7.

Related Functions: CELL, CELLTEXT, CHOOSE, HLOOKUP, INDIRECT, and VLOOKUP.

INDIRECT

Syntax: INDIRECT(string)

Returns: Number or string

Definition: Use INDIRECT to return the contents of the worksheet cell specified by string. Any expression that returns a cell range as text can be used as the argument. This function is not recommended for use in cell formulas. Consider using HLOOKUP, VLOOKUP, or CHOOSE instead. Combining INDIRECT with MAKECELL returns the contents of a variable cell reference.

Example	Result
INDIRECT("A2")	125
INDIRECT(MAKECELL(2, 1))	January

Chapter 3: Alphabetical List of Functions

In the first example, A2 contains 125. In the second example, MAKECELL returns B1 (the first cell in the second column of the worksheet). INDIRECT then returns the cell contents, January.

Related Functions: CELL, CELLTEXT, INDEX, MAKECELL, MAKERANGE, RANGE, and REFERENCE.

INPUTCELL1

Syntax: INPUTCELL1()

Returns: Cell as a string

Definition: Use INPUTCELL1 to return the worksheet cell defined as the first input cell in a data table.

Related Functions: INPUTCELL2 and TABLERANGE.

INPUTCELL2

Syntax: INPUTCELL2()

Returns: Cell as a string

Definition: Use INPUTCELL2 to return the worksheet cell defined as the second input cell in a data table.

Related Functions: INPUTCELL1 and TABLERANGE.

INT

Syntax: INT(number)

Returns: Integer

Chapter 3: Alphabetical List of Functions

Definition: Use INT to return only the integer portion of the specified number. Results are not rounded.

Example	Result
INT(410.98)	410
INT(-2.5)	-2

Related Functions: MOD and ROUND.

INTERCEPT

Syntax: INTERCEPT(known_y's, known_x's)

Returns: Number

Definition: This function calculates the point at which a line will intersect the y-axis by using existing x-values and y-values. The intercept point is based on a best-fit regression line plotted through the known x-values and known y-values. Use the intercept when you want to determine the value of the dependent variable when the independent variable is 0. Known_y's is the dependent set of data. Known_x's is the independent set of data

	A	B	C	D	E
1	2	3	9	1	8
2	6	5	11	7	5

Example	Result
INTERCEPT(A1..E1, A2..E2)	0.0483871

INTEREST

Syntax: INTEREST(principal, payment, term)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: Use INTEREST to calculate the interest rate paid on a principal amount, based on a given payment amount, and a number of payments (term).

Make sure the "time periods" used in payment and term match. They should both be months, years, or days, as appropriate. For example, if the payment is \$100 per month and the term is 5 years, multiply the payment by 12 to equal the number of months in a year (INTEREST(4500, 100*12, 5)). If the payment is annual (one payment per year), you do not need to multiply payment by 12.

Example

INTEREST(20000, 502.46 * 12, 4)
INTEREST(150000, 1230 * 12, 30)

Result

0.079 (7.9% percent)
0.091 (9.1% percent)

In the first example, an automobile valued at \$20,000 is financed for 4 years with monthly payments of \$502.46. The returned annual interest rate is 7.9%. In the second example, a \$150,000 loan with a 30-year term and monthly payments of \$1230 return an annual interest rate of 9.1%.

Related Functions: IRR, LOANTERM, PMT, and PRINCIPAL.

INTRATE

Syntax: INTRATE(settlement, maturity, investment, redemption, basis)

Returns: Number

Definition: This function returns the interest rate for a fully invested security. Settlement is the settlement date. Maturity is the maturity date. Investment is the amount invested. Redemption is the amount to be received at maturity. Basis is the type of day count basis to use.

Chapter 3: Alphabetical List of Functions

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual 365
4	European 30/360

Example	Result
INTRATE("2/15/93", "5/15/93", 1000000, 1014420, 2)	0.058328

INVORDER

Syntax: INVORDER(cost, demand, carry_cost)

Returns: Number

Definition: This function returns the economic order quantity for an inventory item based on order cost, annual demand, and inventory annual carrying cost per unit.

Example	Result
INVORDER(1300, 560, 24.96)	242.00

INVSAFETYSTOCK

Syntax: INVSAFETYSTOCK(ac_demand, stndev_demand, risk)

Returns: Number

Definition: This function returns the stock that should be on hand when a new order arrives for an inventory item based on average demand, standard deviation of demand, and risk of a stock out.

Example	Result
INVSAFETYSTOCK(10.8, 2.7, 0.05)	4.00

Chapter 3: Alphabetical List of Functions

INVSERVICELEVEL

Syntax: INVSERVICELEVEL(quantity, av_demand, stndev_demand, reorder)

Returns: Number

Definition: This function returns the percentage of customer orders that will be satisfied for an inventory item based on order quantity, average demand, standard deviation of demand, and reorder point.

Example

INVSERVICELEVEL(242, 10.8, 2.7, 11)

Result

99.60%

IPMT

Syntax: IPMT(rate, per, nper, pv, fv, type)

Returns: Number

Definition: This function returns the interest payment for a given period for an investment based on periodic, constant payments and a constant interest rate. Rate is the interest rate. Per is the period for which you want to find the interest. Nper is the number of periods. Pv is the present value. Fv is the future value. Type indicates when payments are due.

Set type equal to	If payments are due
0	At the end of the period
1	At the beginning of the period

The following formula calculates the interest due in the last year of a three-year \$8000 loan at 10% annual interest, where payments are made yearly.

Example

IPMT(0.1,3,3,8000)

Result

-\$292.45

Chapter 3: Alphabetical List of Functions

IRR

Syntax: IRR(initial_guess, cash_flow)

Returns: Number

Definition: Use IRR to calculate the approximate internal rate of return on an investment and cash flow series. The calculation is based on repeated approximations for the internal rate of return, starting with the initial_guess.

The internal rate of return is the rate at which the present value of an expected cash flow series equals the value of the initial investment.

initial_guess is your estimate of the internal rate of return and can be any number. Cash_flow is a numeric list of projected future cash flow amounts in the order they are paid out or received. Positive numbers represent inflows of cash and negative numbers represent outflows. The first item in the cash flow represents an initial investment and must be negative. All blank or empty cells referenced by cash_flow are interpreted as 0.

Example	Result
IRR(0.10, -100000, 0, 25000, 40000, 40000)	.02

The example shows that an initial investment of \$100,000 is expected to increase second year revenues by \$25,000 and the remaining two years by \$40,000 each year. The initial_guess is 10%. The IRR result of .02 indicates an approximate internal rate of return of 2% on the \$100,000 over the four years.

Related Functions: INTEREST, NPV, PMT, PRINCIPAL, RATE, and TERM.

ISBLANK

Syntax: ISBLANK(range)

Returns: 0 or 1

Definition: Use ISBLANK to determine if a worksheet cell contains information. Range can be a single cell or a range of cells. If the first cell in range is blank, the function returns 1; otherwise, it returns 0.

Blank cells are empty cells that contain formatting. Both empty and blank cells are considered blank by this function.

Example	Result
ISBLANK(A1..B10)	0
ISBLANK(B7)	1

In the first example, A1 contains text; in the second example, B7 is blank.

Related Functions: ISNUMBER, ISRANGE, and ISSTRING.

ISERR

Syntax: ISERR(expression)

Returns: 0 or 1

Definition: Use ISBLANK to determine whether a function will return an error based on the given arguments. If it will return an error, ISERR returns 1; otherwise, it returns 0.

Example	Result
IF(ISERR(RANGE("A1..D10")), "error", "not an error")	not an error

Chapter 3: Alphabetical List of Functions

In the example, an IF function is used to determine whether the expression argument supplied in the RANGE function is an error. The worksheet range reference in expression is valid, therefore ISERR returns 0 or, in this case, “not an error.”

Related Functions: ERR and ERRNR.

ISNA

Syntax: ISNA(expression)

Returns: 0 or 1

Definition: Use ISNA to return 1 if expression is a N/A (not available) value, and 0 if it is not. The NA function is used to return the N/A status. You can use ISNA with NA to test your worksheet for valid information.

Example	Result
ISNA(f2)	1
ISNA(IF(1 < 0, NA(), 0))	0

In the first example, F2 contains the function NA. In the second example, since the condition to the IF function is false, the result is 0 and the ISNA function returns 0.

Related Function: NA.

ISNUMBER

Syntax: ISNUMBER(expression)

Returns: 0 or 1

Definition: Use ISNUMBER to determine if the elements specified in expression are numbers. The function returns 1 if expression is a number; otherwise, it returns 0.

Chapter 3: Alphabetical List of Functions

Example	Result
ISNUMBER(345.83)	1
ISNUMBER(N44)	0

In the first example, ISNUMBER returns 1 because the specified expression is a number. In the second example, cell N44 contains the string July, which is not a number, causing ISNUMBER to return 0.

Related Functions: ISBLANK, ISRANGE, and ISSTRING.

ISRANGE

Syntax: ISRANGE(expression)

Returns: 0 or 1

Definition: Use ISRANGE to determine if expression is a range. This function returns 1 if expression is a range; otherwise, it returns 0.

Example	Result
ISRANGE(RANGE(MAKERANGE(1,1, 3,3)))	1

In the example above, the RANGE function converts the string returned by the MAKERANGE function into a valid range, causing ISRANGE to return 1.

Related Functions: ISBLANK, ISNUMBER, and ISSTRING.

ISSTRING

Syntax: ISSTRING(expression)

Returns: 0 or 1

Definition: Use ISSTRING to determine if expression is a string. This function returns 1 if expression is a string; otherwise, it returns 0.

Chapter 3: Alphabetical List of Functions

Example	Result
ISSTRING(482)	0
ISSTRING("March")	1
ISSTRING("0.332")	1

In the first example, expression is a number, so ISSTRING returns 0. In the second and third examples, 1 is returned because the argument in each example is a string.

Related Functions: ISBLANK, ISNUMBER, and ISRANGE.

KURT

Syntax: KURT(number1, number2, ...)

Returns: Number

Definition: This function returns the kurtosis of a data set. Kurtosis characterizes the relative peakness or flatness of a distribution compared with the normal distribution. Positive kurtosis indicates a relatively peaked distribution. Negative kurtosis indicates a relatively flat distribution. Number1, number2, ... are 1 to 50 arguments for which you want to calculate kurtosis.

Example	Result
KURT(3,4,5,2,3,4,5,6,4,7)	-0.1518

LARGE

Syntax: LARGE(range, n)

Returns: Number

Definition: This function returns the nth largest value in a data set. Range is the range of data for which you want to determine the nth largest value. N is the position in the cell range of data to return.

Chapter 3: Alphabetical List of Functions

	A	B	C	D	E	F	G	H	I	J
1										

Example	Result
LARGE(A1..J1,3)	5

LCM

Syntax: LCM(number1, number2, ...)

Returns: Number

Definition: This function returns the least common multiple of integers. Number1, number2, ... are 1 to 50 values.

Example	Result
LCM(5,2)	10
LCM(24,36)	72

LEFT

Syntax: LEFT(string, number)

Returns: String

Definition: Use LEFT to retrieve characters from the given string based on the number of characters specified. These characters are retrieved starting at the left side of string until the desired number of characters is met.

Example	Result
LEFT("How are you?", 3)	How
LEFT(D26, 6)	Region

Chapter 3: Alphabetical List of Functions

In the first example, LEFT returns the first three characters of the string. In the second example, cell D26 contains the string, Region A, so LEFT returns the first six characters of the string.

Related Functions: LENGTH, MATCH, MID, and RIGHT.

LENGTH

Syntax: LENGTH(string)

Returns: Number

Definition: Use LENGTH to return the total number of character positions in string.

Example	Result
LENGTH("Lenexa, Kansas")	14
LENGTH(C22)	3

The examples return the number of characters in the specified string. The second example returns the number of characters found in the string entered in cell C22.

Related Functions: LEFT, MID, and RIGHT.

LINEST

Syntax: LINEST(known_y's, known_x's, const, stats)

Returns: Range

Definition: This function calculates the statistics for a line by using "least squares" method to calculate a straight line that best fits your data, and returns a range that describes the line. Known_y's is the set of y-values. Known_x's is an optional set of x-values. Const specifies whether to force the constant b to equal 0. If const is TRUE or omitted, b is calculated normally. If const is FALSE, b is set equal to 0 and the m-values are adjusted to fit $y = mx$. Stats specifies whether to return

Chapter 3: Alphabetical List of Functions

additional statistics. If stats is TRUE, LINEST returns the additional regression statistics. If stats is FALSE or omitted, LINEST returns only the m-coefficients and the constant b.

	A	B	C	D
1	1	9	5	7
2	0	4	2	3

Example

LINEST(A1..D1, A2..D2)

Result

The cells 2,1

LN

Syntax: LN(number)

Returns: Number

Definition: Use LN to calculate the natural logarithm (base e) of number. Number can be any number greater than 0.

Example

LN(1)
LN(E())

Result

0
1

The first example returns 0, the base e value of 1. In the second example, the E function returns the value of the constant e (2.718282). LN then returns 1, the base e logarithm of 2.718282.

Related Functions: E, EXP, LOG, and LOGN.

LOANTERM

Syntax: LOANTERM(principal, payment, interest)

Returns: Number

Definition: Use LOANTERM to return the length of time over which fixed payments must be made to equal the specified principal at the given

Chapter 3: Alphabetical List of Functions

interest rate. Principal, payment, and interest (per payment period) can be any non-negative number.

Example	Result
LOANTERM(5000, 166.07, .12/ 12)	36.00
LOANTERM(12000, 300, .10/ 12)	48.86

In the first example, a term of 36 months is required to pay off a \$5,000 loan, at an annual interest rate of 12% making monthly payments of \$166.07. In the second example, a loan of \$12,000, with an interest rate of 10%, and monthly payments of \$300 is paid off in 48.86 months.

Related Functions: CTERM, INTEREST, PMT, and PRINCIPAL.

LOG

Syntax: LOG(number)

Returns: Number

Definition: Use LOG to calculate the common logarithm (base 10) of number. Number can be any number greater than 0.

Example	Result
LOG(1)	0
LOG(10)	1

The examples return the base 10 logarithms of 1 and 10.

Related Functions: E, LN, and LOGN.

LOGEST

Syntax: LOGEST(known_y's, known_x's, const, stats)

Returns: Range

Chapter 3: Alphabetical List of Functions

Definition: This function calculates an exponential curve that fits your data and returns a range of values that describes the curve. Known_y's is the set of y-values. Known_x's is an optional set of x- values. Const specifies whether to force the constant b to equal 0. If const is TRUE or omitted, b is calculated normally. If const is FALSE, b is set equal to 0 and the m-values are adjusted to fit $y = m^x$. Stats specifies whether to return additional statistics. If stats is TRUE, LOGEST returns the additional regression statistics. If stats is FALSE or omitted, LOGEST returns only the m-coefficients and the constant b.

LOGINV

Syntax: LOGINV(probability, mean, standard_dev)

Returns: Number

Definition: This function returns the inverse of the lognormal cumulative distribution function of x, where $\ln(x)$ is normally distributed with parameters mean and standard_dev. Probability is a probability associated with the lognormal distribution. Mean is the mean of $\ln(x)$. Standard_dev is the standard deviation of $\ln(x)$.

Example	Result
LOGINV(0.039,3.5,1.2)	3.995

LOGN

Syntax: LOGN(number, logbase)

Returns: Number

Definition: Use LOGN to calculate the logarithm of number with the specified logbase and return an exponent. Both attributes of the argument can be any number greater than 0.

Chapter 3: Alphabetical List of Functions

Example	Result
LOGN(8, 2)	3
LOGN(100, 10)	2

In the first example, LOGN determines the exponential element of the equation $2x = 8$. The exponent 3 is returned because 2 to the third power equals 8. In the second example, 10 to the second power equals 100, so LOGN returns the exponent 2.

Related Functions: E, LN, and LOG.

LOGNORMDIST

Syntax: LOGNORMDIST(x, mean, standard_dev)

Returns: Number

Definition: This function returns the cumulative lognormal distribution of x, where $\ln(x)$ is normally distributed with parameters mean and standard_dev. X is the value at which to evaluate the function. Mean is the mean of $\ln(x)$. Standard_dev is the standard deviation of $\ln(x)$.

Example	Result
LOGNORMDIST(3,5,1.2)	0.00057683

LOWER

Syntax: LOWER(string)

Returns: String

Definition: Use LOWER to convert all the characters in the specified string to lowercase.

Example	Result
LOWER("ABCD")	abcd
LOWER(A5)	account 546

Chapter 3: Alphabetical List of Functions

In the examples, LOWER converts each character specified in string to lowercase. In the second example, cell A5 contains ACCOUNT 546.

Related Functions: PROPER and UPPER.

MAGENTA

Syntax: MAGENTA()

Returns: 16711935

Definition: Use MAGENTA to return the RGB value for pure magenta.

Related Functions: BLACK, BLUE, CYAN, GREEN, GREY, RED, RGB, WHITE, and YELLOW.

MAKEBOOKCELL

Syntax: MAKEBOOKCELL(sheet, column, row)

Returns: Cell reference

Definition: This function returns a cell reference built from the supplied arguments.

Example	Result
MAKEBOOKCELL("Sheet1", 3, 5)	Sheet1!C5

Related Function: MAKECELL.

MAKEBOOKRANGE

Syntax: MAKEBOOKRANGE(sheet, column1, row1, sheet2, column2, row2)

Returns: Range as string

Chapter 3: Alphabetical List of Functions

Definition: This function returns a range reference built from the supplied arguments.

Example

MAKEBOOKRANGE("Sheet1a", 3, 5, "Sheet3", 5, 7)

Result

Sheet1!C5..Sheet3!E7

Related Function: MAKERANGE.

MAKECELL

Syntax: MAKECELL(column, row)

Returns: Cell reference

Definition: Use MAKECELL to create a string representation of the worksheet cell reference specified by column and row. Column and row can be any number from 1 to 32768.

Example

MAKECELL(1, 2)

MAKECELL(COL() + 3, ROW() - 6)

Result

"A2"

"G11"

These examples assume you are using column letters (A1) to reference cells, as opposed to column numbers (R1C1). In the first example, MAKECELL returns A2, the intersection of column 1 and row 2. In the second example, MAKECELL returns a string containing a reference to a cell 3 columns to the right and 6 rows up from the current cell location (D17). The result is determined by adding 3 columns to the current column (COL) and subtracting 6 rows from the current row (ROW).

Related Functions: COL, INDIRECT, MAKEBOOKCELL, MAKEBOOKRANGE, MAKERANGE, RANGE, REFERENCE, and ROW.

MAKERANGE

Syntax: MAKERANGE(column1, row1, column2, row2)

Returns: Range reference as a string

Definition: Use MAKERANGE to create a string representation of a worksheet range. The top left cell in the range is specified by column1 and row1, and the bottom right cell is specified by column2 and row2.

MAKERANGE does not accept column letters. Arguments can be any number from 1 to 32768. These examples assume you are using column letters (A1) to reference cells as opposed to column numbers (R1C1).

Example	Result
MAKERANGE(1, 2, 5, 6)	"A2..E6"
MAKERANGE(COL(), ROW(), COL() + 6, ROW() + 6)	"C4..I10"

In the first example, MAKERANGE returns the string representation of the worksheet range A2..E6. Column 1 and row 2 represent the top left cell, A2. Column 5 and row 6 represent the bottom right cell, E6. In the second example, the COL and ROW functions return the column and row number of the current cell location (C4). This formula returns the string representation of a range that includes seven rows and seven columns beginning with column 3, row 4 (C4) and ending with column 9, row 10 (I10).

Related Functions: COL, INDIRECT, MAKEBOOKCELL, MAKEBOOKRANGE, MAKECELL, RANGE, REFERENCE, and ROW.

MATCH

Syntax: MATCH(string, search_string, number)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: Use MATCH to locate the position of a specified portion of text (search_string) within string. MATCH searches string, starting at the number character place, for an exact equal to search_string. It returns the position at which the match starts within string.

MATCH is a case-sensitive function. If there is no exact match during the search, this function returns 0. The first character placement of string is always 1.

Example	Result
MATCH("express mail", "mail", 1)	9
MATCH("(800) 555-1212", "555", 6)	7
MATCH("Tom Bombadil", "bomb", 1)	0

In the first example, MATCH begins at character position 1 in express mail and searches for the string mail. A match is found at position 9. The second example finds a match at position 7. In the third example, “bomb” does not match “Bomb” due to the case-sensitivity of this function. Because no exact match is found, 0 is returned.

Related Functions: CONTAINS, FIND, LEFT, MID, REPLACE, and RIGHT.

MAX

Syntax: MAX(expressions)

Returns: Number

Definition: Use MAX to return the maximum value from a list of numeric and string expressions. Blank and empty cells are ignored. Cells that contain strings or reference blank cells have a value of 0. The maximum number of expressions allowed is 50.

Example	Result
MAX(94, "July", 515, 30.2, 106, 515.3, - 999)	515.3
MAX(0, "Q1")	0

Chapter 3: Alphabetical List of Functions

In the first example, MAX returns 515.3, the maximum value found in the supplied expressions. The second example returns 0 because both expressions equal 0.

Related Functions: DMAX, DMIN, and MIN.

MDETERM

Syntax: MDETERM(range)

Returns: Number

Definition: This function returns the matrix determinant of a range. Range is a numeric range with an equal number of rows and columns.

	A	B
1	3	6
2	1	1

Example

MDETERM(A1.. B2)

Result

-3

MDURATION

Syntax: MDURATION(settlement, maturity, coupon, yld, frequency, basis)

Returns: Number

Definition: This function returns the modified duration for a security. Settlement is the settlement date. Maturity is the maturity date. Coupon is the coupon rate. Yld is the annual yield. Frequency is the number of payments per year. For annual payments, frequency = 1; for semiannual payments, frequency = 2; for quarterly payments, frequency = 4. Basis is the type of day count basis to use.

Chapter 3: Alphabetical List of Functions

Basis Day	Count Basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

MDURATION("1/1/86", "1/1/94", 0.08, 0.09, 2,1)

Result

5.73567

MEDIAN

Syntax: MEDIAN(number1, number2, ...)

Returns: Number

Definition: This function returns the median of the given numbers. Number1, number2, ... are 1 to 50 arguments for which you want the median.

Example

MEDIAN(1,2,3,4,5)

Result

3

MID

Syntax: MID(string, position, length)

Returns: String

Definition: Use MID to return the string of the given length, that begins at the given position in the specified string. The first position in a specified string is always 0.

Example

MID("Office Supplies", 7, 5)

MID("(800) 555-1212", 6, 3)

Result

Suppl

555

Chapter 3: Alphabetical List of Functions

In the first example, MID locates position 7 in “Office Supplies.” The characters returned are the 5 characters located in positions 7 through 11. The second example locates position 6 in “(800) 555-1212” and returns 555 (the 3 characters located in positions 6 through 8).

Related Functions: LEFT, LENGTH, MATCH, and RIGHT.

MIN

Syntax: MIN(expressions)

Returns: Number

Definition: Use MIN to return the minimum value from a list of numeric and string expressions. Blank or empty worksheet cells are ignored. Cells that contain strings have a value of 0. The maximum number of expressions allowed is 50.

Example	Result
MIN("May", 42, 8)	0
MIN(30, 88, -24, 4)	-24

In the first example, MIN returns 9 because “May” has a value of 0. The second example returns the minimum value -24.

Related Functions: DMAX, DMIN, and MAX.

MINUTE

Syntax: MINUTE(date_number)

Returns: Number

Definition: Use MINUTE to calculate the number of minutes past the hour expressed by date_number. A serial date number is a number (from 0 to 401768) that indicates the number of days since December 30, 1899.

Chapter 3: Alphabetical List of Functions

Example	Result
MINUTE(0.4269)	14
MINUTE(TIMEVALUE("12:38p"))	38

The first example returns the decimal representation of minutes for the serial date number 0.4269. In the second example, the TIMEVALUE function converts the time string to a serial date number (approximately 0.53). From this, the decimal representation of minutes is determined by the MINUTE function.

Related Functions: HOUR, NOW, SECOND, TIME, and TIMEVALUE.

MINVERSE

Syntax: MINVERSE(range)

Returns: Range

Definition: This function returns the inverse matrix stored in a range. Range is a numeric range with an equal number of rows and columns.

MIRR

Syntax: MIRR(values, finance_rate, reinvest_rate)

Returns: Number

Definition: This function returns the internal rate of return for a series of periodic cash flows. Values is a reference to cells that contain numbers. Finance_rate is the interest rate you pay. Reinvest_rate is the interest rate you receive.

Chapter 3: Alphabetical List of Functions

	A
1	-120000
2	39000
3	30000
4	21000
5	37000
6	46000

Example

MIRR(A1..A6, 0.10, 0.12)

Result

12.61%

MMULT

Syntax: MMULT (range1, range2)

Returns: Range

Definition: This function returns the matrix product of two ranges. Range1, range2 are the ranges you want to multiply.

MOD

Syntax: MOD(number, divisor)

Returns: Number

Definition: Use MOD to return the remainder after number is divided by divisor. The divisor can be any non-zero number.

Example

MOD(15, 2)

MOD(30, 5)

MOD(-47, 9)

Result

1

0

-2

Related Functions: INT and ROUND.

Chapter 3: Alphabetical List of Functions

MODE

Syntax: MODE(number1, number2, ...)

Returns: Number

Definition: This function returns the most frequently occurring value in a range of data. Number1, number2, ... are 1 to 50 arguments for which you want to calculate the mode.

Example	Result
MODE(5.6, 4, 4,3, 2, 4)	4

MONTH

Syntax: MONTH(date_number)

Returns: 1 through 12

Definition: Use MONTH to calculate the decimal month number for the specified date_number. Date_number is a serial date number from 0 to 401768, indicating the number of days since December 30, 1899. A serial data number is a number (from 0 to 401768) that indicates the number of days since December 30, 1899.

Example	Result
MONTH(NOW())	the current month number
MONTH(DATEVALUE("09/11/ 84"))	9

The first example uses NOW to determine today's date and then returns the current month. In the second example, the DATEVALUE function converts the date_string to a serial date number. From this, the decimal representation of September is determined by the MONTH function.

Related Functions: CMONTH, CWEEKDAY, DAY, MONTHNAME, and YEAR.

MONTHNAME

Syntax: MONTHNAME(date_number)

Returns: String

Definition: Use MONTHNAME to return the name of the month in the specified date_number. Date_number is a serial date number from 0 to 401768, indicating the number of days since December 30, 1899. A series date number is a number (from 0 to 401768) that indicates the number of days since December 30, 1899.

Example

MONTHNAME(2)

MONTHNAME(DATEVALUE("09/11/84"))

Result

January

September

In the first example, January is the string representation of the month of the serial date number 2. In the second example, the DATEVALUE function converts date_string to a serial date number. MONTHNAME returns September, the string representation of the serial date number.

Related Functions: CMONTH, CWEEKDAY, DATEVALUE, MONTH, and NOW.

MROUND

Syntax: MROUND(number, multiple)

Returns: Number

Definition: This function returns a number rounded to the desired multiple. Number is the value to round. Multiple is the multiple to which you want to round the number.

Example

MROUND(10, 3)

Result

9

Chapter 3: Alphabetical List of Functions

MULTINOMIAL

Syntax: MULTINOMIAL(number1, number2, ...)

Returns: Number

Definition: This function returns the ratio of the factorial of a sum of values to the product of factorials. Number1, number2, ... are 1 to 50 values for which you want the multinomial.

Example	Result
MULTINOMIAL(2,3,4)	1260

N

Syntax: N(range)

Returns: Number

Definition: Use N to return the numeric contents of the top left cell of the specified worksheet range. The function returns 0 if the cell contains a string or is blank or empty.

Related Function: S.

NA

Syntax: NA()

Returns: N/A

Definition: Use NA to return the value N/A. A common use for this function is to flag a worksheet cell that needs information. All cells dependent upon a cell with the value of N/A also return N/A.

Related Function: ISNA.

NAME

Syntax: NAME()

Returns: String

Definition: Use NAME to return the name defined for the selected worksheet range or object. If no object is selected, the name of the current worksheet is returned. If no worksheet is active, this function returns an error (ERR 13 Missing worksheet). The NAME function is not case-sensitive.

Related Functions: NUMBER, OBJECTLOCATION, OBJECTNAMES, PATHSSHOWN, and TITLE.

NEGBINOMDIST

Syntax: NEGBINOMDIST(number_f, number_s, probability_s)

Returns: Number

Definition: This function returns the negative binomial distribution. NEGBINOMDIST returns the probability that there will be number_f failures before the number_s success, when the constant probability of a success is probability_s. Number_f is the number of failures. Number_s is the threshold number of successes. Probability_s is the probability of a success.

Example	Result
NEGBINOMDIST(10, 5, 0.25)	0.055049

NFORMAT

Syntax: NFORMAT(number, format)

Returns: String

Chapter 3: Alphabetical List of Functions

Definition: Use NFORMAT to format the specified number, using the format argument. The formatted number is returned as a string.

Three formatting categories are available: numeric, attribute, and precision. Each format type has a correlating format value. When more than one format value is used, each is separated with a "+." The sum of the values is passed as the format argument. Precision formats are represented by the values -15 through +15 and must be used with a numeric format value.

You can also supply a custom format string as **format** to create a unique format for number. All standard custom format characters apply. For specific information on custom format string characters, refer to the FORMAT CUSTOM string command.

The available numeric format values are listed in the following table.

Format	Description
0	General
16	Fixed
32	Currency
48	Percent
64	Scientific
80	Date 1
96	Date 2
112	Date 3
128	Date 4
144	Date 5
160	Time 1
176	Time 2
192	Time 3
208	Time 4

The available attribute format values are identified in the following table.

Format	Description
256	Parenthesis
512	Commas
1024	Hide Zeros
2048	Hidden

Chapter 3: Alphabetical List of Functions

Example	Result
NFORMAT(38233, 128)	"09-03-04"
NFORMAT(23456, 32 + 512 + 2)	"\$23,456.00"
NFORMAT(.55, "\@#.00")	"@.55"

The first example returns a date format for the number 38233. The second example combines currency (32), commas (512), and precision (2) to format the number 23456. The third example formats the number 0.55 using a custom format string.

Related Functions: ADATE, ATIME, CUSTOMFORMAT, CUSTOMFORMATS, DISPLAYLENGTH, and FORMAT.

NOMINAL

Syntax: NOMINAL(effect_rate, npery)

Returns: Number

Definition: This function returns the nominal annual interest rate, given the effective rate and the number of compounding periods per year. Effect_rate is the effective interest rate. Npery is the number of compounding periods per year.

Example	Result
NOMINAL(0.053543,4)	0.0525

NORMAL

Syntax: NORMAL(number)

Returns: Number

Definition: Use NORMAL to return a number selected randomly from a normal distribution, with number representing the standard deviation of the distribution. The mean of the distribution is 0.

Related Functions: EXPONENTIAL, RAND, and UNIFORM.

Chapter 3: Alphabetical List of Functions

NORMDIST

Syntax: NORMDIST(x, mean, standard_dev, cumulative)

Returns: Number

Definition: This function returns the standard normal cumulative distribution function. X is the value for which you want the distribution.

Example	Result
NORMDIST(42, 40, 1.5, 1)	0.908789

NORMINV

Syntax: NORMINV(probability, mean, standard_dev)

Returns: Number

Definition: This function returns the inverse of the normal cumulative distribution for the specified mean and standard deviation. Probability is a probability that corresponds to the normal distribution. Mean is the arithmetic mean. Standard_dev is the standard deviation.

Example	Result
NORMINV(0.908789,40,1.5)	42

NORMSDIST

Syntax: NORMSDIST(probability)

Returns: Number

Definition: This function returns the inverse of the standard cumulative distribution. Probability is a probability corresponding to the normal distribution.

Example	Result
NORMSDIST(1.333333)	0.908789

NORMSINV

Syntax: NORMSINV(probability)

Returns: Number

Definition: This function returns the inverse of the standard normal cumulative distribution. Probability is a probability that corresponds to the normal distribution.

Example	Result
NORMSINV(0.908789)	1.3333

NOW

Syntax: NOW()

Returns: Serial date number

Definition: Use NOW to return a serial date number representing the current date and time. The integer portion of the number represents the number of days that have elapsed since December 30, 1899. The fractional portion represents the time as a fraction of a 24-hour day. A serial date number is a number from 0 to 401768 that indicates the number of days since December 1899. The decimal portion of the number is a fraction of one day. The result of NOW can be formatted as a date or time.

Related Functions: ADATE, ATIME, DATE, DATEVALUE, HOUR, MINUTE, MONTHNAME, SECOND, TIME, TIMEVALUE, and YEAR.

NPER

Syntax: NPER(rate, pmt, pv, fv, type)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the number of periods for an investment based on periodic, constant payments, and a constant interest rate. Rate is the interest rate per period. Pmt is the payment made each period. Pv is the present value. Fv is the future value. Type indicates when payments are due.

Set Type Equal To	If Payments are Due
0	At the end of the period
1	At the beginning of the period

Example	Result
NPER(0.01, -100, -1000, 10000, 1)	60

NPV

Syntax: NPV(interest, income_flow)

Returns: Number

Definition: Use NPV to calculate the net present value of a series of cash flows assuming a fixed periodic interest rate. The periodic interest rate can be any number. Income_flow is a range containing a numeric list of income amounts in the order they are received. The first amount in the income_flow represents the initial investment and must be negative.

Example	Result
NPV(0.1, -5000, 0, 10000, 10000, 10000)	16007.04

In the example, the periodic interest rate is 10% annually with an initial investment of \$5,000, followed by an income flow of \$0, \$10000, \$10000, and \$10000. The net present value of the investment is \$16,007.04.

Related Functions: CTERM, FV, FVL, IRR, PV, PVL, RATE, and TERM.

NRDIMENSIONS

Syntax: NRDIMENSIONS(variable)

Returns: 0, 1, 2, or 3

Definition: Use NRDIMENSIONS to return the number of dimensions in the variable range. If variable is not a range, this function returns 0.

Related Functions: NRELEMENTS.

NRELEMENTS

Syntax: NRELEMENTS(variable, [dimension])

Returns: Number

Definition: Use NRELEMENTS to return the number of elements in the specified dimension of the variable range. If variable is not a range, this function returns 0. To return the total number of cells in a range, do not specify the optional dimension argument.

Related Functions: NRDIMENSIONS.

NRSHEETNAMEDRANGES

Syntax: NRSHEETNAMEDRANGES()

Returns: Number

Definition: This function returns the number of named ranges that are local to the sheet.

Related Functions: RANGENAMES, NRNAMEDRANGES, NRSHEETS, and SHEETRANGENAMES.

Chapter 3: Alphabetical List of Functions

NRSHEETS

Syntax: NRSHEETS()

Returns: Number

Definition: This function returns the number of worksheets in the current workbook, regardless of whether a worksheet contains data.

Related Functions: NRNAMEDRANGES, NRSHEETNAMEDRANGES, RANGENAMES, and SHEETRANGENAMES.

OCT2BIN

Syntax: OCT2BIN(number, places)

Returns: Binary number as a string

Definition: This function converts an octal number to binary. Number is the octal number you want to convert. Places is the number of characters to use.

Example	Result
OCT2BIN(3, 3)	011

Related Functions: BIN2DEC, BIN2HEX, BIN2OCT, DEC2BIN, DEC2HEX, DEC2OCT, HEX2BIN, HEX2DEC, HEX2OCT, OCT2DEC, and OCT2HEX.

OCT2DEC

Syntax: OCT2DEC(number)

Returns: Decimal number

Definition: This function converts an octal number to decimal. Number is the octal number you want to convert.

Chapter 3: Alphabetical List of Functions

Example	Result
OCT2DEC(54)	44

Related Functions: BIN2DEC, BIN2HEX, BIN2OCT, DEC2BIN, DEC2HEX, DEC2OCT, HEX2BIN, HEX2DEC, HEX2OCT, OCT2BIN, and OCT2HEX.

OCT2HEX

Syntax: OCT2HEX(number, places)

Returns: Hexadecimal number as a string

Definition: This function converts an octal number to hexadecimal. Number is the octal number you want to convert. Places is the number of characters to use.

Example	Result
OCT2HEX(100, 4)	0040

Related Functions: BIN2DEC, BIN2HEX, BIN2OCT, DEC2BIN, DEC2HEX, DEC2OCT, HEX2BIN, HEX2DEC, HEX2OCT, OCT2BIN, and OCT2DEC.

ODD

Syntax: ODD(number)

Returns: Number

Definition: This function returns number, rounded up to the nearest odd integer. Number is the value to round.

Example	Result
ODD(1.5)	3
ODD(-1)	-1

Chapter 3: Alphabetical List of Functions

ODDFPRICE

Syntax: ODDFPRICE(settlement, maturity, issue, first, rate, yld, redemption, freq, basis)

Returns: Number

Definition: This function returns the price of a security having an odd first period. Settlement, maturity, issue, and first are dates. Rate is the interest rate. Yld is the yield. Redemption is the redemption value. Freq is the number of payments. For annual payments, frequency = 1; for semiannual, frequency = 2; for quarterly, frequency = 4. Basis is the type of daycount basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

ODDFPRICE("11/11/86", "3/1/99", "10/15/86", "3/1/87", 0.0785, 0.0625, 100, 2,1)

Result

113.597717

ODDLYIELD

Syntax: ODDLYIELD(settlement, maturity, last, rate, pr, redemption, freq, basis)

Returns: Number

Definition: This function returns the yield of a security having an odd last period. Settlement, maturity, and last are dates. Rate is the interest rate. Pr is the price. Redemption is the redemption value. Freq is the number of payments. For annual payments, frequency = 1; for semiannual,

Chapter 3: Alphabetical List of Functions

frequency = 2; for quarterly, frequency = 4. Basis is the type of day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

ODDLYIELD("4/20/87", "6/15/87", "12/24/86", 0.0375, 99.875, 100, 2,0)

Result

0.045192

PEARSON

Syntax: PEARSON(range1, range2)

Returns: Number

Definition: This function returns the Pearson product moment correlation coefficient, r , a dimensionless index that ranges from -1.0 to 1.0 inclusive and reflects the extent of a linear relationship between two data sets. Range1 is set of independent values. Range2 is a set of dependent values.

	A	B	C	D	E
1	9	7	5	3	1
2	10	6	1	5	3

Example

PEARSON(A1..E1, A2..E2)

Result

0.699379

PERCENTILE

Syntax: PERCENTILE(range, n)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the nth percentile of values in a range. Range is the range of data that defines relative standing. N is the percentile value in the range 0...1, inclusive.

	A	B	C	D
1	1	2	3	4

Example

PERCENTILE(A1..D1, 0.3)

Result

1.9

PERCENTRANK

Syntax: PERCENTRANK(range, x, significance)

Returns: Number

Definition: This function returns the rank of a value in a data set as a percentage. Range is the range that defines relative standing. X is the value for which you want to know the rank. Significance is an optional number of significant digits.

	A	B	C	D	E	F	G	H	I	J
1	1	2	3	4	5	6	7	8	9	10

Example

PERCENTRANK(A1..J1,4)

Result

0.333

PERMUT

Syntax: PERMUT(number, number_chosen)

Returns: Number

Definition: This function returns the number of permutations. Number is an integer number of objects. Number_chosen is an integer number of objects in each permutation.

Example

PERMUT(100,3)

Result

970,200

Chapter 3: Alphabetical List of Functions

PFALPHABETA

Syntax: PFALPHABETA(indexes, stocks)

Returns: Range

Definition: This function returns the alpha and beta for equities based on ranges of index series and stock series.

Example

PFALPHABETA(ix_arr, st_arr)

Result

Resultrange(cell1) 105.04

Ix_arr is a range that contains four values: 103.25, 105.50, 104.875, 105.75.

St_arr is a range that contains four values: 168.57, 169.14, 170.21, 170.68.

PFAVGRETURN

Syntax: PFAVGRETURN(contributions, endbalances, eval_period)

Returns: Number

Definition: This function returns average periodic rate of return for a portfolio based on two correlating ranges of net contributions and ending balances, and the number of evaluation periods per year.

Example

PFAVGRETURN(contr_arr, end_arr, 1)

Result

1.19%

Contr_arr is a range that contains four values: 980000, 1050000, 950000,

-100000. End_arr is a range that contains four values: 1000000, 2000000, 3000000, 3000000.

PFCALLVALUE

Syntax: PFCALLVALUE(price, call_price, exp_time, discount_rate, variance)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the value of call options based on stock price, call exercise price, time to expiration, discount rate of return, and variance of stock price.

Example	Result
PFCALLVALUE(30, 32.5, 0.203, 0.03375, 0.0324)	0.25

PFCMPXINTRETURN

Syntax: PFCMPXINTRETURN(contributions, endbalances, dates)

Returns: Number

Definition: This function calculates the dollar-weighted return for a portfolio with random cash flows based on three correlating ranges of net contributions, ending balances, and period begin dates.

Example	Result
PFCMPXINTRETURN(cont_arr, end_arr, date_arr)	16.91%

Cont_arr is a range that contains 12 values: 10000, 2000, 2000, 2000, 1000, 2000, 2000, 1000, 1000, 2000, 5000, 0. End_arr is a range that contains 12 values: 9982, 12102, 14561, 16287, 17331, 19698, 21845, 22963, 24096, 26134, 33001, 0. Date_arr is a range that contains 12 values: DATE(93,1,16), DATE(93,2,3), DATE(93,3,9), DATE(93,4,24), DATE(93,5,11), DATE(93,7,2), DATE(93,8,5), DATE(93,9,19), DATE(93,10,12), DATE(93,11,2), DATE(93,12,4), DATE(93,12,31).

PFCMPXWGHTRETURN

Syntax: PFCMPXWGHTRETURN(contributions, endbalances, dates)

Returns: Number

Definition: This function calculates the time-weighted return for a portfolio with random cash flows based on three correlating ranges of net contributions, ending balances, and period begin dates.

Chapter 3: Alphabetical List of Functions

Example	Result
PFCMPXWGHTRETURN(cont_arr, end_arr, date_arr)	13.61%

PFINTRETURN

Syntax: PFINTRETURN(contributions, endbalances, eval_period)

Returns: Number

Definition: This function returns the dollar-weighted return for a portfolio based on two correlating ranges of net contributions and ending balances, and the number of evaluation periods per year.

Example	Result
PFINTRETURN(cont_arr, end_arr, 1)	1.33%

Cont_arr is a range that contains four values: 980000, 1050000, 950000, -100000. End_arr is a range that contains four values: 100000, 200000, 3000000, 3000000.

PFREGRESSION

Syntax: PFREGRESSION(indexes, stocks)

Returns: Range

Definition: This function returns the correlation coefficient and standard error for equities based on a range of index series and a range of stock series.

Example	Result
PFREGRESSION(ix_arr, st_arr)	Resultrange(cell1) 0.8595 Resultrange(cell2) 0.4258

Ix_arr is a range that contains five values: 311.85, 312.6, 309.14, 307.57, 310.49. St_arr is a range that contains five values: 37.125, 37.0, 35.5, 35.875, 36.75.

Chapter 3: Alphabetical List of Functions

PFRETURN

Syntax: PFRETURN(returns, proportions, standard_devs, correlations)

Returns: Range

Definition: This function returns the expected return and standard deviation for a portfolio based on three correlating ranges of expected rate of return, proportion of portfolio by dollar value, and standard deviation of return, and a range of correlation matrix folded row by row.

Example

PFRETURN(ret_arr, pr_arr, std_arr, corr_arr)

Result

Resultrange(cell1) 0.0975
Resultrange(cell2) 0.0895

Ret_arr is a range with three values: .12, .09, .06. Pr_arr is a range with three values: .5, .25, .25. Std_arr is a range with three values: .17, .09, .035. Corr_arr is a range with three values: .12, -.23, .2.

PFSTNDEVRETURN

Syntax: PFSTNDEVRETURN(contributions, endbalances, eval_period)

Returns: Number

Definition: This function returns the standard deviation for a portfolio based on two correlating ranges of net contributions and ending balances, and the number of evaluation periods per year.

Example

PFSTNDEVRETURN(cont_arr, end_arr, 1)

Result

2.533%

Cont_arr is a range with four values: 980000, 1050000, 950000, -100000.
End_arr is a range with four values: 1000000, 2000000, 3000000, 3000000.

PFWGHTRETURN

Syntax: PFWGHTRETURN(contributions, endbalances, eval_period)

Returns: Number

Definition: This function returns the time-weighted rate of return for a portfolio based on two correlating ranges of net contributions and ending balances, and the number evaluation periods per year.

Example	Result
PFWGHTRETURN(cont_arr, end_arr, 1)	1.16%

Cont_arr is a range that contains four values: 980000, 1050000, 950000, -100000. End_arr is a range that contains four values: 1000000, 2000000, 3000000, 3000000.

PFYIELD

Syntax: PFYIELD(yld, price, years, dividend_growth)

Returns: Number

Definition: This function returns the estimated equity yield based on current stock yield, current stock price, years in projection, and projected annual dividend growth.

Example	Result
PFYIELD(.0625, 32, 5, .15)	17.67%

PI

Syntax: PI()

Returns: Number

Definition: Use PI to return the value of pi (approximately 3.14159).

Related Functions: ASIN, ATAN, and ATAN2.

Chapter 3: Alphabetical List of Functions

PMT

Syntax: PMT(principal, interest, term)

Returns: Number

Definition: Use PMT to calculate the required periodic payment given the principal, interest rate, and term. Make certain the "time periods" used in interest and term match. For example, if the interest rate is 12% per year and the term is 36 months, divide the interest by 12 (months/year). This adjusts the interest time period from .12 per year to .12 / 12 per month, matching the term time period (for example, $\text{PMT}(4516, .12 / 12, 36) = 150.00$).

Example	Result
$\text{PMT}(5000, 0.12 / 12, 36)$	166.07
$\text{PMT}(75000, 0.105 / 12, 30 * 12)$	686.05

The first example determines that monthly payments of \$166.07 are needed for 36 months to pay off a \$5,000 loan at an annual interest rate of 12%. The second example shows that a monthly payment of \$686.05 is required for 30 years to pay off a \$75,000 mortgage at an annual interest rate of 10.5%.

Related Functions: INTEREST, IRR, LOANTERM, and PRINCIPAL.

POISSON

Syntax: POISSON(x, mean, cumulative)

Returns: Number

Definition: This function returns the Poisson distribution. X is the number of events. Mean is the expected numeric value. Cumulative is a logical value that determines the form of the probability distribution returned.

Example	Result
$\text{POISSON}(2, 5, \text{FALSE}())$	0.084224

Chapter 3: Alphabetical List of Functions

POWER

Syntax: POWER(number, power)

Returns: Number

Definition: This function returns the result of a number raised to a power. Number is the base number. Power is the exponent to which the base is raised.

Example	Result
POWER(5,2)	25

PPMT

Syntax: PPMT(rate, per, nper, pv, fv, type)

Returns: Number

Definition: This function returns the payment on the principal for a given period. Rate is the interest rate. Per specifies the period. Nper is the total number of payments. Pv is the present value. Fv is the future value. Type indicates when payments are due.

Set type equal to	If payments are due
0	At the end of the period
1	At the beginning of the period

Example	Result
PPMT(0.8, 10, 12, 200000)	-\$27458.55

PRICE

Syntax: PRICE(settlement, maturity, rate, yld, redemption, freq, basis)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the payment on the principal for a given period. Settlement and maturity are dates. Rate is the coupon rate. Yld is the annual yield. Redemption is the redemption value. Freq is the number of payments. Basis is the type of day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

PRICE("2/15/93", "11/15/99", 0.0575, 0.065, 100, 2, 0)

Result

95.94263694

PRICEDISC

Syntax: PRICEDISC(settlement, maturity, discount, redemption, basis)

Returns: Number

Definition: This function returns the price of a discounted security. Settlement is the settlement date. Maturity is the maturity date. Discount is the discount rate. Redemption is the redemption value. Basis is the type of day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

PRICEDISC("2/15/93", "3/1/93", 0.0525, 100, 2)

Result

99.79583

Chapter 3: Alphabetical List of Functions

PRICEMAT

Syntax: PRICEMAT(settlement, maturity, issue, rate, yld, basis)

Returns: Number

Definition: This function returns the price of a discounted security. Settlement is the settlement date. Maturity is the maturity date. Issue is the discount rate. Rate is the interest rate. Yld is the yield. Basis is the type of day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

PRICEMAT("2/15/93", "4/13/93", "11/11/92", 0.061, 0.061, 0)

Result

99.98449888

PRINCIPAL

Syntax: PRINCIPAL(payment, interest, term)

Returns: Number

Definition: Use PRINCIPAL to return the principal amount paid given the fixed periodic payment and interest rate over the specified term. Make certain the “time periods” used in interest and term match. For example, if the interest rate is 12% per year and the term is 36 months, divide the interest by 12 (months/year). This adjusts the interest time period from .12 per year to .12 / 12 per month, matching the term time period (e.g., PRINCIPAL(166.07,.12 / 12, 36) = 4999.95).

Example

PRINCIPAL(166.07, 0.12 / 12, 36)

PRINCIPAL(686.05, 0.105 / 12, 30 * 12)

Result

4999.95

74999.51

Chapter 3: Alphabetical List of Functions

In the first example, if monthly payments of \$166.07 are made for 36 months at a 12% monthly interest rate, \$4999.95 is the principal amount paid at the end of the 36 months. In the second example, if monthly payments of \$686.05 are made for 30 years, with a 10.5% annual interest rate, \$74,999.51 is the principal amount paid at the end of 30 years.

Related Functions: INTEREST, IRR, LOANTERM, and PMT.

PROB

Syntax: PROB(x_range, prob_range, lower_limit, upper_limit)

Returns: Number

Definition: This function returns the probability that values are between two limits. X_range is the range of numeric values of x. Prob_range is a set of probabilities associated with values in x_range. Lower_limit is the lower bound. Upper_limit is the optional upper bound.

	A	B	C	D
1	0	1	2	3
2	0	0	0	0

Example

PROB(A1..D1, A2..D2, 1, 3)

Result

0.8

PROBZSCORE

Syntax: PROBZSCORE(prob)

Returns: Number

Definition: This function obtains the Z score from a probability.

Example

PROBZSCORE(0.1)

Result

-1.28

PRODUCT

Syntax: PRODUCT(number1, number2, ...)

Returns: Number

Definition: This function multiplies all the numbers given as arguments and returns the product. Number1, number2, ... are 1 to 50 values that you want to multiple. If cells A2..C2 contain 5, 15, and 30, PRODUCT(A2..C2) = 2250.

PROPER

Syntax: PROPER(string)

Returns: String

Definition: Use PROPER to convert the first character of each word in string to uppercase.

Example

PROPER("joseph r. williams")

Result

Joseph R. Williams

Related Functions: LOWER and UPPER.

PV

Syntax: PV(payment, interest, term)

Returns: String

Definition: Use PV to return the present value of an investment given the fixed periodic payment and interest rate (per period) over the specified term. Make sure the "time periods" used in payment and interest match. For example, if the payment is \$1000 per month and the interest rate is .0915 annually, divide the interest by 12 (months/year). This adjusts the interest time period from .0915 per year to .0915 / 12 per month,

Chapter 3: Alphabetical List of Functions

matching the payment time period (i.e., $PV(1000, .0915 / 12, 5) = 4887.63$).

Example	Result
$PV(166.07, 0.12 / 12, 36)$	4999.95
$PV(686.05, 0.105 / 12, 30 * 12)$	74999.51

The first example shows that when monthly payments of \$166.07 are made at a monthly interest rate of 12%, at the end of a 36 month period the total investment equals \$4,999.95. The second example shows that when \$686.05 per month is invested for 30 years with an annual interest rate of 10.5%, the amount accumulated is \$74,999.51.

Related Functions: CTERM, FV, FVL, NPV, PVL, RATE, and TERM.

PVDEFPAYVALUE

Syntax: PVDEFPAYVALUE(disc_rate, no_pays, pay_period, no_periods, pres_value, speriod_opt, comp_period)

Returns: Number

Definition: This function returns the payment in a deferred series based on the discount rate, number of payments, pay period, number of deferred periods, present value, short period option (0 = compound, 1 = simple), and compounding periods per year.

Example	Result
$PVDEFPAYVALUE(0.065, 180, 12, 120, 100000, 0, 12)$	1656.74

PVDEFVALUE

Syntax: PVDEFVALUE(disc_rate, no_pays, pay_period, no_periods, amount, speriod_opt, comp_period)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the net present value of a deferred annuity based on the discount rate, number of payments, payments per year, number of deferred periods, payment amount, short period option (0 = compound, 1 = simple), and compounding periods per year.

Example	Result
PVDEFVALUE(0.06, 240, 12, 468, 241.67, 0, 1)	3538.38

PVDISCOUNTRATE

Syntax: PVDISCOUNTRATE(pres_value, pay_period, amount, no_pays, comp_period, speriod_opt, timing)

Returns: Number

Definition: This function returns the annual rate of return required to provide a fixed cash stream from a specified base. The calculation is based on present value, payments per year, payment amount, number of payments, compounding periods per year, short period option (0 = compound, 1 = simple) and number of payment timing (0 = end of period, 1 = beginning of period).

Example	Result
PVDISCOUNTRATE(20000, 12, 200, 216, 365, 0, 0)	9.96%

PVL

Syntax: PVL(payment, interest, term)

Returns: Number

Definition: Use PVL to calculate the present value of a lump sum payment due in the future based on a specified periodic interest rate over a given term. Make sure the "time periods" used in payment and interest match. For example, if the payment is \$1000 per month and the interest rate is .0915 annually, divide the interest by 12 (months/year). This adjusts the interest time period from .0915 per year to .0915 / 12 per

Chapter 3: Alphabetical List of Functions

month, matching the payment time period (i.e., $PVL(1000, .0915 / 12, 5) = 962.73$).

Example	Result
<code>PVL(5000, 0.10, 5)</code>	3104.61

In the example, if you elect to receive a payoff today (in lieu of a lump sum payoff of \$5,000 in 5 years accruing an annual interest rate of 10%), the amount you should receive is \$3,104.61.

Related Functions: CTERM, FV, FVL, NPV, PV, RATE, and TERM.

PVPAYAMOUNT

Syntax: `PVPAYAMOUNT(comp_period, pay_period, no_pays, interest, pres_value, speriod_opt, timing)`

Returns: Number

Definition: This function returns a constant periodic payment to amortize a given beginning balance based on compounding periods per year, payments per year, number of payments, interest rate, present value, short period option (0 = compound, 1 = simple), and payment timing (0 = end of period, 1 = beginning of period).

Example	Result
<code>PVPAYAMOUNT(12, 12, 360, 0.1025, 200000, 0, 0)</code>	1792.20

PVPAYMENTS

Syntax: `PVPAYMENTS(pres_value, pay_period, amount, interest, comp_period, speriod_opt, timing)`

Returns: Number

Definition: This function returns the number of payments of a given amount required to amortize a debt or available from an annuity based on

Chapter 3: Alphabetical List of Functions

present value, payments per year, payment amount interest rate, compounding periods per year, short period option (0 = compound, 1 = simple), and payment timing (0 = end of period, 1 = beginning of period).

Example	Result
PVPAYMENTS(42670,12,800,0.08,12, 0,0)	66.00

PVPRESENTVALUE

Syntax: PVPRESENTVALUE(fut_value, interest, no_periods, speriod_opt, comp_period)

Returns: Number

Definition: This function returns the present value of a future lump sum payment based on future value interest rate, compounding periods before payment, short period option (0 = compound, 1 = simple), and compounding periods per year.

Example	Result
PVPRESENTVALUE(10000, 0.08, 45, 1, 365)	9901.87

PVSCHEDULE

Syntax: PVSCHEDULE(comp_period, pay_period, no_pays, interest, pres_value, timing, amount, speriod_opt)

Returns: Range

Definition: This function returns a range with interest, principal, and balance left of an amortization schedule based on compounding periods per year, payments per year, number of payments interest rate, present value, payment timing (0 = end of period, 1 = beginning of period), payment value, and short period option (0 = compound, 1 = simple).

Chapter 3: Alphabetical List of Functions

Example

PVSCHEDULE(12, 12, 30*12, .1025, 200000, 0, 1792.20, 0)

Result

Resultrange(cells1,1) 1708.33

Resultrange(cells1,2) 83.87

Resultrange(cells1,3) 199916.13

...

Resultrange(cells360,1) 15.23

Resultrange(cells360,2) 1783.23

Resultrange(cells350,3) 0

PVVALUE

Syntax: PVVALUE(comp_period, pay_period, no_payments, interest, amount, speriod_opt, timing)

Returns: Number

Definition: This function returns the amount that can be amortized by a series of fixed payments based on compounding periods per year, payments per year, number of payments, interest rate, payment amount, short period option (0 = compound, 1 = simple), and payment timing (0 = end of period, 1 = beginning of period).

Example

PVVALUE(12, 12, 36, 0.073, 356, 0, 0)

PVVALUE(1, 12, 240, 0.085, 3000, 1, 0)

Result

11478.52

353561.54

QUARTILE

Syntax: QUARTILE(range, quart)

Returns: Number

Definition: This function returns the quartile of a data set. Range is the range of numeric values for which you want the quartile. Quart indicates which value to return.

Chapter 3: Alphabetical List of Functions

If quart equals	QUARTILE returns
0	Minimum value
1	First quartile (25th percentile)
2	Median value (50th percentile)
3	Third quartile (75th percentile)
4	Maximum value

A screenshot of an Excel spreadsheet grid. The columns are labeled A through H, and the rows are labeled 1 through 12. The grid is shaded in a light gray color.

Example QUARTILE(A1..H1, 1)	Result 3.5
---------------------------------------	----------------------

QUOTIENT

Syntax: QUOTIENT(numerator, denominator)

Returns: Number

Definition: This function returns the integer portion of a division. Numerator is the dividend. Denominator is the divisor.

Example QUOTIENT(5,2)	Result 2
---------------------------------	--------------------

RADIANS

Syntax: RADIANS(degrees)

Returns: Number in radians

Definition: Use RADIANS to convert a number of degrees to radians.

Example RADIANS(90) RADIANS(A3)	Result 1.5708 -0.7854
--	------------------------------------

Chapter 3: Alphabetical List of Functions

In the first example, the value of 90 degrees is converted to 1.5708 radians. In the second example, A3 contains the value -45 .

Related Function: DEGREES.

RAND

Syntax: RAND()

Returns: Number from 0 through 1

Definition: Use RAND to return a random number selected randomly from a uniform distribution between 0 and 1.

Related Functions: EXPONENTIAL, NORMAL, and UNIFORM.

RANDBETWEEN

Syntax: RANDBETWEEN(bottom, top)

Returns: Number

Definition: This function returns a random number between the numbers you specify. Bottom is the smallest integer RANDBETWEEN will return. Top is the largest integer RANDBETWEEN will return.

Example

RANDBETWEEN(5, 10)

Possible Result

5.966521195

RANGE

Syntax: RANGE(string)

Returns: Range

Chapter 3: Alphabetical List of Functions

Definition: Use RANGE to convert the supplied string to a worksheet range. This function is used in formulas to reference a valid string. String must convert to a valid cell or range.

Example	Result
SUM(RANGE("A1..A5"))	15
SUM(RANGE(MAKERANGE(1, 1, 1, 5)))	15

In the examples, range A1..A5 contains the values 1, 2, 3, 4, 5. In the first example, the range reference exists as a string until it is converted by the RANGE function. The sum of the contents in the range is then obtained by the SUM function. In the second example, the MAKERANGE function returns the string representing the range reference "A1..A5." The RANGE function then converts the string to a range to be used as an argument for the SUM function.

Related Functions: INDIRECT, MAKEBOOKCELL, MAKEBOOKRANGE, MAKECELL, MAKERANGE, REFERENCE, SAMERANGE, and SHEETWORKAREA.

RANGENAME

Syntax: RANGENAME(range)

Returns: String

Definition: Use RANGENAME to return the name of the specified worksheet range if a name has been defined for it. If the range is not named, null is returned. Range names are sequentially numbered in order of definition. Therefore, you can specify a number for the range argument.

Related Functions: NAMEDRANGE, NRNAMEDRANGES, NRSHEETNAMEDRANGES, RANGENAMED, RANGENAMES, SHEETRANGENAMES, and VALIDRANGENAME.

Chapter 3: Alphabetical List of Functions

RANGENAMED

Syntax: RANGENAMED(name)

Returns: Range

Definition: Use RANGENAMED to return the range of the current worksheet range name. If name is not an existing range name, ERR 14 is returned.

Example

REFERENCE(RANGENAMED("xxx"))

Result

Sheet1!\$A\$1..\$1\$13

Related Functions: NAMEDRANGE, NRNAMEDRANGES, NRSHEETNAMEDRANGES, RANGENAME, RANGENAMES, and VALIDRANGENAME.

RANGENAMES

Syntax: RANGENAMES()

Returns: Range of strings

Definition: Use RANGENAMES to return a range of all the named ranges defined for the current workbook. If no named ranges have been defined, null is returned.

Related Functions: NAMEDRANGE, NRNAMEDRANGES, NRSHEETNAMEDRANGES, OBJECTNAMES, RANGENAME, RANGENAMED, SHEETRANGENAMES, TEXTFONTS, and VALIDRANGENAME.

RANGEPART

Syntax: RANGEPART(range, selector)

Returns: String

Chapter 3: Alphabetical List of Functions

Definition: This function returns the following information for the selected range based on the following selectors.

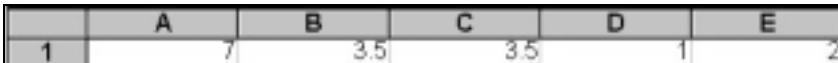
Selector Value	Returns
0	Book name
1	Left column number
2	Top row number
3	Right column number
4	Bottom row number
5	First sheet name
6	Last sheet name
Other	Argument Is Out Of Range error

RANK

Syntax: RANK(number, ref, order)

Returns: Number

Definition: This function returns the rank of a number in a list of number. Number is the number whose rank you want to find. Ref is a range of numbers. Order is a number specifying how to rank number.



	A	B	C	D	E
1	7	35	35	1	2

Example

RANK(A1, A1..E1,1)

Result

5

RATE

Syntax: RATE(future_value, present_value, term)

Returns: Number

Definition: Use RATE to calculate the interest rate for an annuity, where present_value increases to a specified future_value when compounded over the given term.

Chapter 3: Alphabetical List of Functions

Example	Result
RATE(20000, 5000, 10 * 12)	0.0116

In the example, an interest rate of 1.16% per month is required for an investment of \$5,000 to grow to \$20,000 in 120 months (10 years).

Related Functions: CTERM, FV, FVL, IRR, NPV, PV, PVL, and TERM.

RECEIVED

Syntax: RECEIVED(settlement, maturity, investment, discount, basis)

Returns: Number

Definition: This function returns the amount received at maturity for a fully invested security. Settlement is the settlement date. Maturity is the maturity date. Investment is the amount invested. Discount is the discount rate. Basis is the type of day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example	Result
RECEIVED("2/15/93", "5/15/93", 1000000, 0.0572, 2)	1014343.95

RED

Syntax: RED()

Returns: 255

Definition: Use RED to return the RGB value for pure red.

Related Functions: BLACK, BLUE, CYAN, GREEN, GREY, MAGENTA, RGB, WHITE, and YELLOW.

REFERENCE

Syntax: REFERENCE(range)

Returns: Range as a string

Definition: Use REFERENCE to convert the specified worksheet range to a string representing the range. The reference is converted to an absolute reference.

Example

REFERENCE(RANGENAMED("xxx"))

Result

\$A\$1..\$B\$5

The example uses REFERENCE in conjunction with SELECTION to determine the selected worksheet range (A1..B5).

Related Functions: INDIRECT, MAKECELL, MAKERANGE, RANGE, and SELECTION.

RELATEDBOOKS

Syntax: RELATEDBOOKS()

Returns: Range of strings

Definition: Use RELATEDBOOKS to return a range describing all the workbooks referenced by the target workbook. Information about each referenced workbook is contained in a separate row of the range. The first element in each row contains the referenced workbook name and the second element contains the path to the workbook.

Related Functions: OPENRELATEDBOOKS and REMEMBERLINKEDDATA.

Chapter 3: Alphabetical List of Functions

REPEAT

Syntax: REPEAT(string, number)

Returns: String

Definition: Use REPEAT to copy string the specified number of times.

Example	Result
REPEAT("*", 6)	*****
REPEAT("X", 3)	XXX
REPEAT("ab", 2)	abab

REPLACE

Syntax: REPLACE(string1, position, length, string2)

Returns: String

Definition: Use REPLACE to replace a portion of one string with another. The function locates string1, moves to the specified position in string1, and replaces the portion of it specified in length with string2. The first character place of either string is position 0. If position is greater than the number of characters in string1, string2 is appended to the end of the original text.

Example	Result
REPLACE(B7, 14, 9, "best")	JSheet is the best spreadsheet
REPLACE("Month End Results", 0, 5, "Year")	Year End Results

In the first example, B7 (string1) contains “JSheet is the graphical spreadsheet.” The replacement begins at position 14 (g) for a length of 9 (graphical). As shown, best replaces graphical. In the second example, the replacement begins at position 0 (M) for a length of 5 characters (Month). These characters are replaced by Year (string2).

Related Functions: CONTAINS, FIND, and MATCH.

Chapter 3: Alphabetical List of Functions

RGB

Syntax: RGB(red, green, blue)

Returns: RGB value

Definition: Use RGB to return the value for a color with the given red, green, and blue intensities. The value of each color intensity ranges from 0 (darkest) to 255 (lightest).

Example

RGB(255, 255, 255)

Result

16777215 (value of white)

Related Functions: BLACK, BLUE, BRIGHTNESS, CYAN, GREEN, GREY, HSBTORGB, HUE, MAGENTA, RED, SATURATION, WHITE, and YELLOW.

RIGHT

Syntax: RIGHT(string, number)

Returns: String

Definition: Use RGB to retrieve the rightmost number of characters from the given string. If number is larger than the length of string, string is returned. The first character position in string is always 1.

Example

RIGHT("07/23/92", 2)

RIGHT(C10, 4)

Result

92

7900

The first example returns the year (92) from 07/23/92. In the second example, C10 contains Mountain View, CA 94039-7900. RIGHT returns the last four digits of the zip code (7900).

Related Functions: LEFT, LENGTH, MATCH, and MID.

Chapter 3: Alphabetical List of Functions

ROLLOVERYEAR

Syntax: ROLLOVERYEAR()

Returns: Number

Definition: This function returns the current rollover year.

ROMAN

Syntax: ROMAN(number, form)

Returns: String

Definition: This function converts an Arabic numeral to roman, as text. Number is the Arabic numeral you want converted. Form is the number specifying the type of roman numeral you want.

Form	Type
0	Classic
1	More concise
2	More concise
3	More concise
4	Simplified

Example	Result
ROMAN(499, 0)	CDXCIX
ROMAN(499, 1)	LDVLIV
ROMAN(499, 2)	XDIX
ROMAN(499, 3)	VDIV
ROMAN(499, 4)	ID

ROUND

Syntax: ROUND(number, precision)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: Use ROUND to return number rounded at the specified decimal place, precision. When a number is rounded to a negative precision, all of the values to the right of the decimal are dropped and the absolute number of significant digits (specified with precision) are replaced with zeroes.

Example	Result
ROUND(1.2945, 1)	1.3
ROUND(-4.5678, 2)	-4.57
ROUND(12945, -3)	13000

These examples round number to the specified precision. By using a negative precision, the third example rounds 12945 to the thousands decimal place.

Related Functions: INT and MOD.

ROUNDDOWN

Syntax: ROUNDDOWN(number, num_digits)

Returns: Number

Definition: This function rounds a number down, toward zero. Number is the number you want to round down. Num_digits specifies the number of digits to which you want to round number.

Example	Result
ROUNDDOWN(3.2, 0)	3
ROUNDDOWN(3.14159, 3)	3.141

ROUNDUP

Syntax: ROUNDUP(number, num_digits)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function rounds a number up, away from zero. Number is the number you want to round up. Num_digits specifies the number of digits to which you want to round number.

Example	Result
ROUNDUP(3.2, 0)	4
ROUNDUP(3.14159, 3)	3.142

ROW

Syntax: ROW()

Returns: Number

Definition: Use ROW to return the row number of the current cell.

Related Functions: COL, COLOF, COLS, MAKECELL, MAKERANGE, ROWOF, and ROWS.

ROWOF

Syntax: ROWOF(range)

Returns: Number

Definition: Use ROWOF to return the row number of the first row in the specified range.

Example	Result
ROWOF(A1)	1
ROWOF(E3..J4)	3

The specified worksheet range in the first example represents column 1 and row 1, therefore ROWOF returns 1. The second example returns the first row of the range.

Related Functions: COL, COLOF, COLS, ROW, ROWS, and BOOKOF.

Chapter 3: Alphabetical List of Functions

ROWS

Syntax: ROWS(range)

Returns: Number

Definition: Use ROWS to return the number of rows in the specified range.

Example	Result
ROWS(A1..A5)	5
ROWS(A1)	1

The first example returns 5, the number of rows in the specified range. The second example returns 1 as the number of rows in the range.

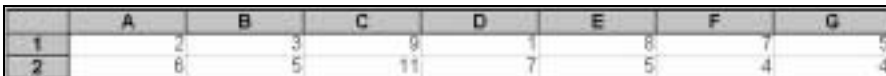
Related Functions: COL, COLOF, COLS, ROW, ROWOF, and BOOKOF.

RSQ

Syntax: RSQ(known_y's, known_x's)

Returns: Number

Definition: This function returns the square of the Pearson product moment correlation coefficient through data points in known_y's and known_x's. Known_y's and known_x's are ranges of data points.



The screenshot shows a spreadsheet with two rows of data. Row 1 contains values 2, 3, 6, 1, 8, 7, 5. Row 2 contains values 6, 5, 11, 7, 5, 4, 4. The columns are labeled A through G.

	A	B	C	D	E	F	G
1	2	3	6	1	8	7	5
2	6	5	11	7	5	4	4

Example	Result
RSQ(A1..G1, A2..G2)	0.05795

Chapter 3: Alphabetical List of Functions

S

Syntax: S(range)

Returns: String

Definition: Use S to return the string stored in the top left cell of the specified range. If this cell contains a number or is blank or empty, null is returned.

Example

S(A1..D8)
S(F7..H9)

Result

Monday
""

In the first example, A1 contains “Monday.” In the second example, F7 contains the number 458 and null is returned.

Related Function: N.

SAMERANGE

Syntax: SAMERANGE(range1, range2)

Returns: Number

Definition: Use this function to determine whether the provided ranges are the same. The function returns 1 if the ranges are the same, 0 if the ranges are not the same. In the examples below, the range A1..A5 has been named “test.”

Example

SAMERANGE("A1..A5", "A1..A5")
SAMERANGE("A1..A5", "A1..A2")
SAMERANGE("test", "A1..A5")
SAMERANGE("test", "A1..A2")
SAMERANGE("R1C1..R5C1", "A1..A5")

Result

1
0
1
0

SATURATION

Syntax: SATURATION(rgb_value)

Returns: Number

Definition: Use SATURATION to return the saturation of the given RGB value. In a method similar to specifying the red, green, and blue values of a color (RGB), you can also specify the same color using hue, saturation, and brightness. This function translates a RGB value into the saturation component of the HSB color.

Related Functions: BRIGHTNESS, HSBTORGB, HUE, and RGB.

SECOND

Syntax: SECOND(date_number)

Returns: Number in seconds

Definition: Use SECOND to extract the second from the specified serial date_number. The fractional component of the serial date_number represents the time as a fraction of a 24-hour day. A serial date number is a number (from 0 to 401768) that indicates the number of days since December 30, 1899. The decimal portion of the number is a fraction of one day.

Example	Result
SECOND(32330.2876)	8
SECOND(TIMEVALUE(B6))	9

The first example converts the seconds component of the serial date number to the decimal representation. In the second example, the TIMEVALUE function converts the string contents of B6 (12:38:09p) to a serial date number. From this, the decimal representation of the seconds component (9) is returned by the SECOND function.

Related Functions: HOUR, MINUTE, NOW, TIME, and TIMEVALUE.

Chapter 3: Alphabetical List of Functions

SERIESSUM

Syntax: SERIESSUM(x, n, m, coefficients)

Returns: Number

Definition: This function returns the sum of a power series. X is the input value. N is the initial power to which you want to raise x. M is the step by which to increase n. Coefficients is a set of coefficients.

	A	B	C	D
1	1.00	-0.5	0.041666667	-0.001388888889

Example

SERIESSUM(0.78539825, 0, 2, A1..D1)

Result

0.7071031537

SHEET

Syntax: SHEET()

Returns: String

Definition: This function returns the current worksheet name in the current workbook.

Related Function: NAME.

SHEETCELLS

Syntax: SHEETCELLS(sheet)

Returns: Number

Definition: This function returns the number of active cells in a worksheet. If sheet is a string, the function returns the values for the

Chapter 3: Alphabetical List of Functions

sheet with that name. If sheet is a number, the function returns the values for the worksheet that is displayed at that position in the window. If there is no worksheet, an error (ERR 13) is returned. If the given worksheet is invalid, the ERR 12 is returned.

Related Functions: CELLS, FORMULAS, and SHEETFORMULAS.

SHEETFORMULAS

Syntax: SHEETFORMULAS(sheet)

Returns: Number

Definition: This function returns the number of formulas in a worksheet. If sheet is a string, the values for the sheet with that name are returned. If sheet is a number, the values for the worksheet displayed at that position in the window are returned. If there is no worksheet, ERR 13 is returned. If the given worksheet is invalid, ERR 12 is returned.

Related Functions: CELLS, FORMULAS, and SHEETCELLS.

SHEETLIST

Syntax: SHEETLIST()

Returns: Range of strings

Definition: This function returns a range of worksheet names in the current workbook.

Related Function: SHEET.

Chapter 3: Alphabetical List of Functions

SHEETNAMEOF

Syntax: SHEETNAMEOF(range)

Returns: String

Definition: This function returns the sheet name only portion of the specified range.

Example

SHEETNAMEOF(A1..B5)

Result

Sheet1

SHEETRANGE

Syntax: SHEETRANGE()

Returns: Range

Definition: Use SHEETRANGE to return a range that encloses the currently used portion of the worksheet. The upper left area of the returned range always begins on worksheet cell A1.

SHEETRANGENAMES

Syntax: SHEETRANGENAMES(sheetname)

Returns: Range of strings

Definition: This function returns a range of the named ranges that are local to the worksheet. The function requires a sheetname as an argument. This function does not pick up any 3D ranges that may encompass the given worksheet.

Related Function: RANGENAMES.

SHEETS

Syntax: SHEETS(range)

Returns: Number

Definition: This function returns the number of sheets in a specified range.

SIGN

Syntax: SIGN(number)

Returns: -1, 0, or 1

Definition: Use SIGN to return a value representing the sign used for the specified number. The following values are returned by this function.

Sign of number	Returns
None (zero)	0
Positive	1
Negative	-1

Example	Result
SIGN(88.13)	1
SIGN(-300)	-1
SIGN(ABS(-300))	1

The first example returns 1, because 88.13 is a positive number. In the second example, -300 causes SIGN to return the negative sign values. In the third example, the ABS function returns the absolute value of -300. The SIGN function then returns the positive sign value of 300 (the absolute value).

Related Functions: ABS and FACTORIAL.

Chapter 3: Alphabetical List of Functions

SIN

Syntax: SIN(number)

Returns: -1 to 1

Definition: Use SIN to calculate the sine of a number. Number can be any number given in radians.

Example	Result
SIN(0.1)	0.10
SIN(C9-1.7)	0.96

The first example returns the sine of 0.1. In the second example, 1.7 is subtracted from the value in cell C9, which contains 2.99. SIN then calculates the sine of the resulting value.

Related Functions: ACOS, ACOSH, ASIN, ASINH, ATAN, ATAN2, ATANH, COS, COSH, SINH, TAN, and TANH.

SINH

Syntax: SINH(number)

Returns: Number

Definition: Use SINH to return the hyperbolic sine of a number.

Example	Result
SINH(0.1)	0.10
SINH(A12/2)	10.02

The first example returns the hyperbolic sine of 0.1. In the second example, A12 contains 6, which is divided by 2. SINH then converts the resulting 3 to the hyperbolic sine 10.02.

Related Functions: ACOS, ACOSH, ASIN, ASINH, ATAN, ATAN2, ATANH, COS, COSH, SIN, TAN, and TANH.

Chapter 3: Alphabetical List of Functions

SKEW

Syntax: SKEW(number1, number2, ...)

Returns: Number

Definition: This function returns the skewness of a distribution. Number1, number2, ... are 1 to 50 arguments for which you want to calculate skewness.

Example	Result
SKEW(3,4,5,2,3,4,5,6,4,7)	0.359543

SLN

Syntax: SLN(original_cost, salvage_value, life)

Returns: -1 to 1

Definition: Use SLN to compute the straight-line depreciation of an asset's value for one period based on the original_cost, salvage_value, and useful life. Unlike the accelerated method of depreciation, the straight-line method has the same amount of depreciation for each year. The salvage_value represents the value of an asset at the end of the period of useful life. Life represents the number of periods the asset depreciates.

Example	Result
SLN(10000, 1000, 7)	1285.71
SLN(B5, B6, B7)	2200.00

In the first example, a \$10,000 automobile with a useful life of 7 years and a salvage value of \$1,000 results in a depreciation expense of \$1285.71. The second example shows how worksheet cell references can be used in place of absolute values in this function. Each argument is a reference to a cell that contains an appropriate value.

Related Functions: DDB and SYD.

Chapter 3: Alphabetical List of Functions

SLOPE

Syntax: SLOPE(known_y's, known_x's)

Returns: Number

Definition: This function returns the slope of the linear regression line. Known_y's is a range of numeric dependent data points. Known_x's is the set of independent data points.

	A	B	C	D	E	F	G
1	2	3	9	1	8	7	5
2	6	5	11	7	5	4	4

Example

SLOPE(A1..G1, A2..G2)

Result

.305556

SMALL

Syntax: SMALL(range, n)

Returns: Number

Definition: This function returns the nth smallest value in a data set. Range is the range of data for which you want to determine the nth smallest value. N is the position in the cell range of data to return.

	A	B	C	D	E	F	G	H	I	J
1	3	4	5	2	3	4	5	6	4	7

Example

SMALL(A1..J1, 4)

Result

4

SQRT

Syntax: SQRT(number)

Returns: Number

Definition: Use SQRT to calculate the square root of the specified number. Any non-negative number can be used.

Example	Result
SQRT(16)	4
SQRT(D19)	3.7
SQRT(5*5)	5

The first example returns the square root of 16. In the second example, cell D19 contains the value 13.69. The third example returns the square root of 25.

Related Function: SUMSQ.

SQRTPI

Syntax: SQRTPI(number)

Returns: Number

Definition: This function returns the square root of number multiplied by pi.

Example	Result
SQRTPI(2)	2.506628

STANDARDIZE

Syntax: STANDARDIZE(x, mean, standard_dev)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns a normalized value from a distribution characterized by mean and standard_dev. X is the value you want to normalize. Mean is the arithmetic mean of the distribution. Standard_dev is the standard deviation of the distribution.

Example	Result
STANDARDIZE(42, 40, 1.5)	1.333333

STD

Syntax: STD(expressions)

Returns: Number

Definition: Use STD to calculate the standard deviation of a population for the specified list of expressions. The standard deviation of a population represents the average amount of deviation from the population mean that occurs in a list of numbers. The value returned by STD is equal to the square root of the variance as calculated by the VAR function. You can have up to 50 items in the expressions list. Strings are considered to have a value of 0. Blank and empty cells are ignored.

Example	Result
STD(4.5, 6.7, 27.2, 32.4)	12.26
STD(3, 1176, 27, 543, 999)	482.96

Both examples return the standard deviation of a population for each specified list of numbers.

Related Functions: DSTD, DSTDEV, STDEV, and VAR.

STDEV

Syntax: STDEV(expressions)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: Use STDEV to calculate the standard deviation of a sample for the specified list of expressions. The standard deviation of a sample represents the average amount of deviation from the sample mean that occurs in a list of numbers. The value returned by STDEV is equal to the square root of the variance as calculated by the VARIANCE function. Strings are considered to have a value of 0. Blank and empty cells are ignored.

Example	Result
STDEV(4.5, 6.7, 27.2, 32.4)	14.16
STDEV(D1..D5)	539.97

Both examples return the standard deviation of a sample for each specified list of numbers or values in the specified range.

Related Functions: AVG, DAVG, DSTD, DSTDEV, STD, and VARIANCE.

STDEVP

Syntax: STDEVP(number1, number2, ...)

Returns: Number

Definition: This function calculates a standard deviation based on the entire population given as arguments. The standard deviation is a measure of how widely values are dispersed from the average value (the mean). Number1, number2, ... are 1 to 50 number arguments corresponding to a sample of a population.

Example	Result
STDEVP(4.5, 6.7, 27.2, 32.4)	12.26356

STEYX

Syntax: STEYX(known_y's, known_x's)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the standard error of the predicted y-value for each x in the regression. Known_y's is a range of dependent data points. Known_x's is the set of independent data points.

	A	B	C	D	E	F	G
1	2	3	9	1	8	7	5
2	6	5	11	7	5	4	4

Example

STEYX(A1..G1, A2..G2)

Result

3.305719

STR

Syntax: STR(number)

Returns: String

Definition: Use STR to convert a positive or negative number to a string.

Example

STR(10)
STR(0.68133)

Result

"10"
"0.68133"

Related Functions: EVALUATE, STRING, and VALUE.

STRING

Syntax: STRING(number, precision)

Returns: String

Definition: Use STRING to convert the specified number to a string, with precision representing the displayed number of decimal places.

Chapter 3: Alphabetical List of Functions

Example	Result
STRING(10, 2)	"10.00"

The example returns the number 10 as a string with a precision of 2.

Related Functions: EVALUATE, STR, and VALUE.

SUM

Syntax: SUM(expressions)

Returns: Number

Definition: Use SUM to calculate the sum of all the numbers specified in expressions. Strings have a value of 0. Blank and empty cells are ignored. You can have up to 50 items in expressions.

Example	Result
SUM(1,3,5,7,9)	25
SUM(2 + 2, "18", 3 * 7)	25

Related Functions: AVG, DAVG, and DSUM.

SUMIF

Syntax: SUMIF(range, criteria, sum_range)

Returns: Number

Definition: This function adds the cells specified by a given criteria. Range is the range of cells you want evaluated. Criteria is the criteria that defines which cells will be counted. Sum_range is the actual cells to sum.

Chapter 3: Alphabetical List of Functions

	A	B	C	D
1	100000	200000	300000	400000
2	7000	14000	21000	28000

Example

SUMIF(A1..D1, ">160000", A2..D2)

Result

63000

SUMPRODUCT

Syntax: SUMPRODUCT(range1, range2)

Returns: Number

Definition: This function multiplies corresponding components in the given ranges, and returns the sum of those products. Range1, range2, ... are 2 to 50 ranges whose components you want to multiply and then add.

	A	B	C	D	E
1	3	4		2	7
2	8	6		6	7
3	1	9		5	3

Example

SUMPRODUCT(A1..B3, D1..E3)

Result

156

SUMSQ

Syntax: SUMSQ(expressions)

Returns: Number

Definition: Use SUMSQ to square each number in expressions and then calculate the sum of all the squares. Strings have a value of 0. Blank and empty cells are ignored. You can have up to 50 items in the expressions list.

Chapter 3: Alphabetical List of Functions

Example

SUMSQ(1, 4, 12, 8, 5)

Result

250

The example shows that $1^2 + 4^2 + 12^2 + 8^2 + 5^2$ returns the value 250.

Related Functions: DSUMSQ and SQRT.

SUMX2MY2

Syntax: SUMX2MY2(range_x, range_y)

Returns: Number

Definition: This function returns the sum of the difference of the squares or corresponding values in two ranges. Range_x and range_y are the ranges of values.

	A	B	C	D	E	F	G
1	2	3	9	1	8	7	5
2	6	5	11	7	5	4	4

Example

SUMX2MY2(A1..G1, A2..G2)

Result

-55

SUMX2PY2

Syntax: SUMX2PY2(range_x, range_y)

Returns: Number

Definition: This function returns the sum of the sum of the squares that correspond to values in two ranges. Range_x and range_y are the ranges of values.

Chapter 3: Alphabetical List of Functions

	A	B	C	D	E	F	G
1	2	3	9	1	8	7	5
2	6	5	11	7	5	4	4

Example

SUMX2PY2(A1..G1, A2..G2)

Result

521

SUMXMY2

Syntax: SUMXMY2(range_x, range_y)**Returns:** Number

Definition: This function returns the sum of the squares of differences of corresponding values in two ranges. Range_x and range_y are the ranges of values.

	A	B	C	D	E	F	G
1	2	3	9	1	8	7	5
2	6	5	11	7	5	4	4

Example

SUMXMY2(A1..G1, A2..G2)

Result

79

SYD

Syntax: SYD(cost, salvage, life, period_number)**Returns:** Number

Definition: Use SYD to compute the depreciation of an asset, at an accelerated rate, over a specified period of time. The greatest depreciation occurs early in the useful life of the asset. The depreciable cost is the actual cost less the salvage value. Unlike the straight-line method of depreciation (SLN), SYD uses a variant amount of depreciation

Chapter 3: Alphabetical List of Functions

for each year. Cost (original cost) and salvage (salvage value) can be any number. Life (periods of useful life) and period_number (the period of useful life) can be any positive number.

Example	Result
SYD(15000, 2000, 7, 1)	3250.00 (year one)
SYD(15000, 2000, 7, 3)	2321.43 (year three)
SYD(15000, 2000, 7, 7)	464.29 (year seven)

In the examples, period_number 1, 3, and 7 of a seven-year depreciation schedule are computed. These computations are based on a \$15,000 automobile that has a salvage value of \$2,000 and a useful life of 7 years.

Related Functions: DDB and SLN.

TABLERANGE

Syntax: TABLERANGE()

Returns: Range as string

Definition: Use TABLERANGE to return the current data table range. Data tables are used in one- or two-input variable tables (also called What If tables). They calculate the results of changing one value in one or more formulas, or two values in a single formula.

Related Functions: INPUTCELL1 and INPUTCELL2.

TAN

Syntax: TAN(number)

Returns: Number

Definition: Use TAN to return the tangent of number expressed in radians. Number cannot be a multiple of $\pi/2$.

Chapter 3: Alphabetical List of Functions

Example	Result
TAN(0.12)	0.12
TAN(1)	1.56

Related Functions: ACOS, ACOSH, ASIN, ASINH, ATAN, ATAN2, ATANH, COS, COSH, SIN, SINH, and TANH.

TANH

Syntax: TANH(number)

Returns: Number

Definition: Use TANH to return the hyperbolic tangent of number. The result is between -1 and 1.

Example	Result
TANH(0.12)	0.12
TANH(1)	0.76

Related Functions: ACOS, ACOSH, ASIN, ASINH, ATAN, ATAN2, ATANH, COS, COSH, SIN, SINH, and TAN.

TBILLEQ

Syntax: TBILLEQ(settlement, maturity, discount)

Returns: Number

Definition: This function returns the bond-equivalent yield for a treasury bill. Settlement is the settlement date. Maturity is the maturity date. Discount is the discount rate.

Example	Result
TBILLEQ("3/31/93", "6/1/93", 0.0914)	0.094151

TBILLPRICE

Syntax: TBILLPRICE(settlement, maturity, discount)

Returns: Number

Definition: This function returns the price per \$100 face value for a treasury bill. Settlement is the settlement date. Maturity is the maturity date. Discount is the discount rate.

Example	Result
TBILLPRICE("3/31/93", "6/1/93", 0.09)	98.45

TBILLYIELD

Syntax: TBILLYIELD(settlement, maturity, pr)

Returns: Number

Definition: This function returns the yield for a treasury bill. Settlement is the settlement date. Maturity is the maturity date. Pr is the price.

Example	Result
TBILLYIELD("3/31/93", "6/1/93", 98.45)	9.1417%

TDIST

Syntax: TDIST(x, degrees_freedom, tails)

Returns: Number

Definition: This function returns the Student's t-distribution. X is the value at which you want to evaluate the distribution. Degrees_freedom is the number of degrees of freedom. Tails specifies the number of distribution tails to return.

Example	Result
TDIST(1.96, 60, 2)	0.054645

Chapter 3: Alphabetical List of Functions

TERM

Syntax: TERM(payment, interest, future_value)

Returns: Number

Definition: Use TERM to compute the number of payment periods required for the given payment and interest rate to accrue to the specified future_value. Make sure the "time periods" used in payment and interest match. For example, if the payment is \$100 per month and the annual interest rate is 10.5%, divide the interest by 12 (months/year). This adjusts the interest time period from 10.5% per year to $10.5 / 12$ per month, matching the payment time period (TERM(100, .105 / 12, 500,000)). If the payment is annual (only one payment per year), you do not need to divide interest by 12.

Example	Result
TERM(150, 0.10/12, 20000)	90.04
TERM(200, 0.105/12, 500,000)	359.28

The first example shows that 90.04 monthly investments (approximately 7.5 years) of \$150 each, at an annual interest rate of 10% are required to accumulate a \$20,000 balance at the end of the term. In the second example, if you deposit \$200 a month into an annuity that yields 10.5% annual interest, it takes 359.28 months (approximately 30 years) for the annuity to mature to \$500,000.

Related Functions: CTERM, FV, FVL, IRR, NPV, PV, PVL, and RATE.

TICKER

Syntax: TICKER(value, direction, size)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function enables a scrolling region to be created in the worksheet. Value can be a single value (or function return value), a cell name, a range where one of the dimensions is 1 (such as a vector), or a region that specifies a vector range. Direction is a character string that specifies the scroll direction ("up", "down", "left", or "right"), and size is the number of cells in the scrolling direction. Whenever one of the specified values changes, all previous values scroll into the next cell.

For example, create a scrolling region 15 cells wide in C5..C20, scrolling right to left, and put the following formula in cell C5: =TICKER(A1, "left", 15). The current value of A1 is placed in the right- most cell. Each time A1 changes, all values scroll to the left and the right-most cell is updated with the new value of A1.

TIME

Syntax: TIME(hour, minute, second)

Returns: Serial date number

Definition: Use TIME to return a serial date number corresponding to the time specified by the hour, minute, and second arguments. The serial date number is a decimal number between 0 and 1 that represents time as a fraction of a 24-hour day. A serial date number is a number from 0 to 401768 that indicates the number of days since December 30, 1899.

Example	Result
TIME(11, 42, 55)	0.49

In the example, TIME converts 42 minutes and 55 seconds past the hour of 11 into the serial date number 0.49.

Related Functions: ATIME, DATE, HOUR, MINUTE, NOW, SECOND, and TIMEVALUE.

Chapter 3: Alphabetical List of Functions

TIMEVALUE

Syntax: TIMEVALUE(time_string)

Returns: Serial date number

Definition: Use TIMEVALUE to convert a time that is represented by time_string into a serial date number. TIMEVALUE can recognize and convert times in many different formats (for example, 11:59:00 AM, 11:59:00, and 23:59:00). A serial date number is a number from 0 to 401768 that indicates the number of days since December 30, 1899.

Example	Result
TIMEVALUE("9:27:50")	0.39
TIMEVALUE("21:16:00")	0.89

In both examples, the TIMEVALUE function converts each time_string to the correlating serial date number.

Related Functions: ATIME, DATE, HOUR, MINUTE, NOW, SECOND, and TIME.

TINV

Syntax: TINV(probability, degrees_freedom)

Returns: Number

Definition: This function returns the inverse of the Student's t-distribution. Probability is a probability associated with the two-tailed Student's t-distribution. Degrees_freedom is the number of degrees of freedom.

Example	Result
TINV(0.054645, 60)	1.96

TITLESRANGE

Syntax: TITLESRANGE()

Returns: Range as string

Definition: Use TITLESRANGE to return the worksheet range used to define the titles in the current worksheet.

Related Functions: TITLEGRIDHIDDEN and TITLESHIDDEN.

TREND

Syntax: TREND(known_y's, known_x's, new_x's const)

Returns: Number

Definition: This function returns values along a linear trend. Fits a straight line (using the method of least squares) to the ranges known_y's and known_x's. Returns the y-values along that line for the range of new_x's that you specify. Known_y's is the set of y values. Known_x's is the optional set of x values. New_x's are new x-values. Const specifies whether to force the constant b to equal 0.

TRIM

Syntax: TRIM(string)

Returns: String

Definition: Use TRIM to remove all of the leading and trailing spaces and remove duplicate spaces from between the words in the specified string.

Example

TRIM("one two")

TRIM(" J.R. ") & TRIM("Pims ")

Result

one two

J.R.Pims

Chapter 3: Alphabetical List of Functions

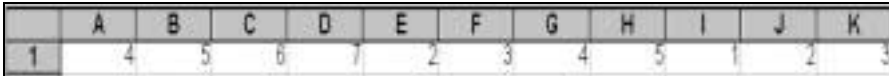
The first example removes the extra spaces between one and two. The second example removes the leading and trailing spaces in the specified strings, then joins the strings and returns the combined result.

TRIMMEAN

Syntax: TRIMMEAN(range, percent)

Returns: Number

Definition: This function returns the mean of the interior of a data set. Range is the range of values to trim and average. Percent is the fractional number of data points to exclude from the calculation.



	A	B	C	D	E	F	G	H	I	J	K
1	4	5	6	7	2	3	4	5	1	2	3

Example

TRIMMEAN(A1..K1, 0.2)

Result

3.77778

TRUE

Syntax: TRUE()

Returns: 1

Definition: Use TRUE to return a value of 1. TRUE can be used in an IF statement to clarify the conditional portion of a function.

Example

IF(A31 = 7, TRUE(), FALSE())

Result

1

In the example, the IF statement determines whether the equation is TRUE or FALSE. In this case, A31 contains a value of 7, so 1 (TRUE) is returned.

Related Functions: FALSE and IF.

Chapter 3: Alphabetical List of Functions

TRUNC

Syntax: TRUNC(number, num_digits)

Returns: Number

Definition: This function truncates a number to an integer by removing the fractional part of the number. Number is the number you want to truncate. Num_digits is a number that specifies the precision of the truncation.

Example

TRUNC(12345.9877, 2)

Result

12345.9800

Related Functions: INT, ROUND, and ROUNDUP.

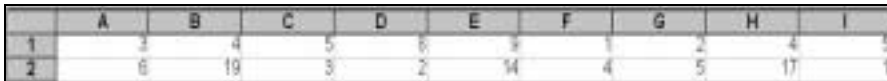
TTEST

Syntax: TTEST(range1, range2, tails, type)

Returns: Number

Definition: This function returns the probability associated with a Student's t-Test. Range1 is the first data set. Range2 is the second data set. Tails specifies the number of distribution tails. Type is the kind of t-Test to perform.

If type equals	This test is performed
1	Paired
2	Two-sample equal variance (homoscedastic)
3	Two-sample unequal variance (heteroscedastic)



	A	B	C	D	E	F	G	H	I
1									
2	6	19	5	2	14	4	5	17	1

Example

TTEST(A1..I1, A2..I2, 2,1)

Result

0.196016

Chapter 3: Alphabetical List of Functions

UNIFORM

Syntax: UNIFORM(number)

Returns: Number

Definition: Use UNIFORM to return a real number selected randomly from a uniform distribution between 0 and number.

Related Functions: EXPONENTIAL, NORMAL, and RAND.

UPPER

Syntax: UPPER(string)

Returns: String

Definition: Use UPPER to convert all of the characters in the string to uppercase.

Example	Result
UPPER("abcd")	ABCD

Related Functions: LOWER and PROPER.

VALUE

Syntax: VALUE(string)

Returns: Number

Definition: Use VALUE to convert string to a number. Conversion proceeds from left to right and stops with the first character that cannot be interpreted as part of a number.

Example	Result
VALUE("2300")	2300.00
VALUE("23a44")	23.00

Chapter 3: Alphabetical List of Functions

In both examples, each specified string is converted to a number. The resulting number indicates the default precision setting, 2. Because the second example cannot interpret the alphabetic character (a), the conversion returns a value of 23.00.

Related Functions: EVALUATE, STR, and STRING.

VAR

Syntax: VAR(expressions)

Returns: Number

Definition: Use VAR to calculate the population variance of the specified expressions. Strings have a value of 0 and blank and empty cells are ignored.

Example	Result
VAR(4.5, 6.7, 27.3, 32.4)	150.87
VAR(3, 1176, 27, 543, 999)	233252.64

Both examples return the population variance for each list of expressions.

Related Functions: DVAR, DVARIANCE, STD, and VARIANCE.

VARIANCE

Syntax: VARIANCE(expressions)

Returns: Number

Definition: Use VARIANCE to calculate the sample variance of the specified expressions. Strings or references to blank cells are considered to have a value of 0. Blank and empty cells are ignored.

Chapter 3: Alphabetical List of Functions

Example	Result
VARIANCE(4.5, 6.7, 27.3, 32.4)	201.16
VARIANCE(3, 1176, 27, 543, 999)	291565.80

Both examples return the sample variance for each list of expressions.

Related Functions: DVAR, DVARIANCE, STDEV, and VAR.

VARP

Syntax: VARP

Returns: VARP(number1, number2, ...)

Definition: This function calculates variance based on the entire population. Number1, number2, ... are 1 to 50 number arguments that correspond to a sample of a population.

VDB

Syntax: VDB(cost, salvage, life, start, end, factor, no_switch)

Returns: Number

Definition: This function returns the depreciation of an asset. Cost is the initial cost. Salvage is the salvage value. Life, start, and end are time periods. Factor is the rate at which the balance declines. No_switch is a logical value.

Example	Result
VDB(2400, 300, 3650, 0, 1)	1.32

VLOOKUP

Syntax: VLOOKUP(search_item, range, offset)

Returns: Number or string

Definition: Use VLOOKUP to return the contents of a worksheet cell position, determined by the given arguments. VLOOKUP searches the left column of the specified range for the search_item. The lookup finds a cell in the left column whose contents are the closest value equal to or less than the search item. It then returns the contents of the cell that is offset the specified number of columns from the found cell. The column number of the found cell is offset 0. If you use an offset of 1, the returned value is the contents of the cell one column to the right of the found cell. The search_item and offset can be any number.

An offset of 0 returns an index number (beginning with 0) of the located item if the search_item is a string. When a numeric search_item is supplied with an offset of 0, the cell contents is returned. If the offset exceeds the range boundaries (range), ERR 25 is returned. If the search_item is a string, the item must match the cell contents exactly. If no match is found, ERR 6 is returned.

Because the function stops the search as soon as it finds a match, numeric data in the search column must be in ascending order. When strings are used in the lookup column, no ordering is required.

The following example uses the income tax table shown in the illustration below as the VLOOKUP table data.

Chapter 3: Alphabetical List of Functions

	A	B	C	D	E
1	Taxable		Married Filing	Married Filing	Head of
2	Income	Single	Jointly	Separately	Household
3	34000 - 34050	6,882	5,107	7,317	5,978
4	34050 - 34100	6,896	5,121	7,331	5,992
5	34100 - 34150	6,910	5,135	7,345	6,006
6	34150 - 34200	6,924	5,149	7,359	6,020
7	34200 - 34250	6,938	5,163	7,373	6,034
8	34250 - 34300	6,952	5,177	7,387	6,048
9	34300 - 34350	6,966	5,191	7,401	6,062
10	34350 - 34400	6,980	5,205	7,415	6,076

Example

VLOOKUP("34150 – 34200", A3..E10, 4)

Result

6020.00

In this example, VLOOKUP determines the withholding taxes on an income between \$34,150 and \$34,200 using Head of Household status. As shown in the first argument ("34150 – 34200"), the income ranges in column A are strings. Unlike numeric search_items, string search_items must match exactly. When an exact match for the search_item is found (A6), the offset of 4 returns the Head of Household withholding.

Related Functions: CHOOSE, HLOOKUP, and INDEX.

WEIBULL

Syntax: WEIBULL(x, alpha, beta, cumulative)

Returns: Number

Definition: This function returns the Weibull distribution. X is the value at which to evaluate the function. Alpha and beta are parameters to the distribution. Cumulative determines the form of the function.

Example

WEIBULL(105, 20, 100, TRUE())

Result

0.929581

Chapter 3: Alphabetical List of Functions

WHITE

Syntax: WHITE()

Returns: 16777215

Definition: Use WHITE to return the RGB value for pure white.

Related Functions: BLACK, BLUE, CYAN, GREEN, GREY, MAGENTA, RED, RGB, and YELLOW.

XIRR

Syntax: XIRR(values, dates, guess)

Returns: Number

Definition: This function returns the internal rate of return for a schedule of cash flows that is not necessarily periodic. Values is a series of cash flows. Dates is a schedule of payment dates. Guess is a number that you guess is close to the result of XIRR.

	A	B	C	D	E
1	-10000	2750	4250	3250	2750
2	33604	33664	33907	34015	34060

Example

XIRR(A1..E1, A2..E2, .1)

Result

.37336

XNPV

Syntax: XNPV(rate, values, dates)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function returns the net present value for a schedule of cash flows that is not necessarily periodic. Rate is the discount rate. Values is a series of cash flows. Dates is a schedule of payment dates.

	A	B	C	D	E
1	-10000	2750	4250	3250	2750
2	33604	33664	33907	34015	34060

Example

XNPV(.09, A1..E1, A2..E2)

Result

2086.647602

YEAR

Syntax: YEAR(date_number)

Returns: Number

Definition: Use YEAR to calculate the decimal year number for the serial date number, date_number. A serial date number is a number (from 0 to 401768) that indicates the number of days since December 30, 1899.

Example

YEAR(34072)

YEAR(DATEVALUE("09/11/84"))

Result

1993

1984

The first example returns the decimal representation of the year for the serial date number 34072. In the second example, the DATEVALUE function converts the date string to a serial date number. From this, the decimal representation of the year (1984) is returned by the YEAR function.

Related Functions: DATE, DATEVALUE, DAY, MONTH, and NOW.

YELLOW

Syntax: YELLOW()

Returns: 65535

Chapter 3: Alphabetical List of Functions

Definition: Use YELLOW to return the RGB value for pure yellow.

Related Functions: BLACK, BLUE, CYAN, GREEN, GREY, MAGENTA, RED, RGB, and WHITE.

YIELD

Syntax: YIELD(settlement, maturity, rate, pr, redemption, frequency, basis)

Returns: Number

Definition: This function returns the yield on a security. Settlement and maturity are dates. Rate is the coupon rate. Pr is the price. Redemption is the redemption value. Frequency is the number of payments. Basis is the type of day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

YIELD("2/15/91", "11/15/99", 0.0575, 95.04287, 100, 2, 0)

Result

0.065

YIELDDISC

Syntax: YIELDDISC(settlement, maturity, pr, redemption, basis)

Returns: Number

Definition: This function returns the annual yield for a discounted security. Settlement and maturity are dates. Pr is the price. Redemption is the redemption value. Basis is the type of day count basis to use.

Chapter 3: Alphabetical List of Functions

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

YIELDDISC("2/15/93", "3/1/93", 99.795, 100,2)

Result

5.2823%

YIELDMAT

Syntax: YIELDMAT(settlement, maturity, issue, rate, pr, basis)

Returns: Number

Definition: This function returns yield on a security. Settlement, maturity, and issue are dates. Rate is the interest rate. Pr is the price. Basis is the type of day count basis to use.

Basis	Day count basis
0	US (NASD) 30/360
1	Actual/actual
2	Actual/260
3	Actual/365
4	European 30/360

Example

YIELDMAT("3/15/93", "11/3/93", "11/8/92", 0.0625, 100.0123, 0)

Result

0.060954

ZSCOREPROB

Syntax: ZSCOREPROB(zscore)

Returns: Number

Chapter 3: Alphabetical List of Functions

Definition: This function obtains the probability from a Z score.

Example

ZSCOREPROB(-1.07692307692308)

Result

0.14

ZTEST

Syntax: ZTEST(range, x, sigma)

Returns: Number

Definition: This function returns the two-tailed P-value of a z-test. Range is the range of data against which to test x. X is the value to test. Sigma is the population (known) standard deviation.

	A	B	C	D	E	F	G	H	I	J
1	3	6	7	8	6	5	4	2	1	9

Example

ZTEST(A1..J1, 4)

Result

0.09057

Chapter 3: Alphabetical List of Functions

INDEX

A

Arguments, function, 1-2
 formulas, 1-5
 number, 1-2
 optional, 1-5
 range, 1-4
 string, 1-3
 types of, 1-2

C

Color functions, 2-8

D

Database functions, 2-6
Date functions, 2-8

E

Engineering functions, 2-5

F

Financial functions, 2-2
Formula operators
 evaluation order, 1-8
 types of, 1-6
Formulas
 as function arguments, 1-5
 evaluation order of operators,
 1-8
 using operators in, 1-6

Functions

 argument types, 1-2
 arguments, 1-2
 color, 2-8
 database, 2-6
 date, 2-8
 engineering, 2-5
 financial, 2-2
 formula arguments, 1-5
 keywords, 1-1
 logical, 2-5
 number arguments, 1-2
 numeric, 2-1
 optional arguments, 1-5
 range arguments, 1-4
 statistical, 2-4
 string arguments, 1-3
 text, 2-7
 time, 2-8
 worksheet, 2-9
 worksheet status, 2-9

K

Keywords, function, 1-1

L

Logical functions, 2-5

M

Multimedia functions group, 2-5

N

Number arguments, 1-2
Numeric functions, 2-1

O

Operators
 evaluation order of, 1-8
 in formulas, 1-6

P

Precedence, operator, 1-8

R

Range arguments, 1-4
Ranges, types of, 1-4

S

Statistical functions, 2-4
String arguments, 1-3

T

Text functions, 2-7
Time functions, 2-8

W

Worksheet functions, 2-9
Worksheet status functions, 2-9