FinalCalc.hyper

Khalid Aldoseri

COLLABORATORS			
	TITLE : FinalCalc.hyper		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Khalid Aldoseri	January 17, 2023	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

FinalCalc.hyper

1.1 Appendix 1: Formula Functions...

```
ABOVE
ABOVE

ABS
ABS(x)

ACOS
ACOS(x)

ACOSH
ACOSH(x)

ACOTH
ACOTH(x)

ALPHA
ALPHA(range)

ASIN
ASIN(x)

ASINH
ASINH(x)

ATAN
ATAN(x)

ATAN2
ATAN2(x,y)

ATANH
ATANH(x)

AVG
AVG(multiple range/cell/expression arguments)

BELOW
BELOW
```

```
BETA
BETA(range)

BIN
BIN(x)

BINTO
BINTO(str)

CASE
CASE(x,v1,v2,v3,....,vn)

CAVG
CAVG(expression,multiple range arguments)

CCOUNT
CCOUNT(expression,multiple range arguments)

CELL
CELL(column,row)  CELL(sheet,col,row)

CFIRST
CFIRST(expression,multiple range arguments)

CHAR
CHAR(x)

CHOOSE
CHOOSE(x,v1,v2,v3,....,vn)

CLAST
CLAST(expression,multiple range arguments)

CLEAN
CLEAN(str)

CMAX
CMAX(expression,multiple range arguments)

CMIN
CMIN(expression,multiple range arguments)

CODE
CODE(str)

COL
COL()  COL(cell or range)  COL(range,cell)

COLOR
COLOR(str)

COLS
COLS(range)

COMB
COMB(n,x)
```

```
CORR
CORR(range)

COS
COS(x)

COSH
COSH(x)

COTAN
COTAN(x)

COTH
COTH(x)

COUNT
COUNT(multiple range/cell/expression arguments)

COVAR
COVAR(range)

CPICK
CPICK(offset,expression,multiple range arguments)

CSUM
CSUM(expression,multiple range arguments)

DATE
DATE(year,month,day)

DATESTR
DATESTR(str,x)

DATEVALUE
DATEVALUE(str)

DAVG
DAVG(offset)

DAY
DAY(date)

DAYS
DAYS(year,month)

DCOUNT
DCOUNT(offset)

DEC
DEC(x)

DEGREES
DEGREES(x)

DMAX
DMAX(offset)
```

DMIN
DMIN(offset)

DSTD
DSTD(offset)

DSUM
DSUM(offset)

DVAR
DVAR(offset)

ERROR
ERROR

EVALUATE
EVALUATE(str)

EXACT
EXACT(str1,str2)

EXP
EXP(x)

FACT
FACT(x)

FALSE
FALSE

FAVG
FAVG(expression,multiple range arguments)

FCOUNT
FCOUNT(expression,multiple range arguments)

FFIRST
FFIRST(expression,multiple range arguments)

FIELD
FIELD

FIND
FIND(str1,str2,x)

FIRST
FIRST(multiple cell/range/expression arguments)

FLAST
FLAST(expression,multiple range arguments)

FMAX
FMAX(expression,multiple range arguments)

FMIN
FMIN(expression,multiple range arguments)

```
FPICK
FPICK(offset,expression,multiple range arguments)

FRAC
FRAC(x)

FSIGN
FSIGN(F value,degrees of freedom 1,degrees of freedom 2)

FSUM
FSUM(expression,multiple range arguments)

FV
FV(present amount,interest rate,periods)

FVA
FVA(annuity,interest rate,periods)

GAMMA
GAMMA(x)

GETCOLOR
GETCOLOR(cell)

GETSTYLE
GETSTYLE(cell)

HERE
HERE  HERE([sheet offset,]col offset,row offset)

HEX
HEX(x)

HEXTO
HEXTO(str)

HLOOK
HLOOK(value to be looked up,range,row offset)

HOUR
HOUR(time)

IF
IF(expression,value if expression true, value if expression false ↩
   )

INDEX
INDEX(range,[sheet offset,]column offset,row offset)

INF
INF

INT
INT(x)

IRR
```

```
IRR(range of cash flows,guess)

ISERR
ISERR(expression)

ISINF
ISINF(expression)

ISNUM
ISNUM(x)

ISRANGE
ISRANGE(range)

ISSTR
ISSTR(x)

IZERR
IZERR(expression)

KEEP
KEEP(str,template)

LAST
LAST(multiple cell/range/expression arguments)

LEFT
LEFT  LEFT(str,x)

LEN
LEN(str)

LIMIT
LIMIT(x,a,b,value if x<a,value if a<=x<=b,value if b<x)

LOG
LOG(x)

LOGN
LOGN(x)

LOWER
LOWER(str)

MAX
MAX(multiple range/cell/expression arguments)

MID
MID(str,start,length)

MIN
MIN(multiple range/cell/expression arguments)

MINUTE
MINUTE(time)

MOD
```

```
MOD(x,y)

MONTH
MONTH(date)

MONTHS
MONTHS(date)

NAMESTR
NAMESTR(cell/range)

NINF
NINF

NOT
NOT(x)

NOW
NOW

NPV
NPV(range of cash flows,discount rate)

NUM
NUM(number or string)

OCT
OCT(x)

OCTTO
OCTTO(str)

PATH
PATH(), PATH(dir/file name)

PATTERN
PATTERN(str)

PERM
PERM(n,x)

PI
PI

PICK
PICK(offset,multiple cell/range/expression arguments)

PMT
PMT(principal,interest,term of loan)

PROD
PROD(multiple range/cell/expression arguments)

PROPER
PROPER(str)

PV
```

```
PV(future amount,interest rate,periods)

PVA
PVA(annuity,interest rate,periods)

RADIANS
RADIANS(x)

RAND
RAND

RANGE
RANGE([sheet,]col,row,[sheet2,]col2,row2)

RATE
RATE(future value,present value,periods)

RATEA
RATEA(future value,present value,periods)

REGRES
REGRES(independant variables range,dependant variables range,item ↩
    ,offset)

REMOVE
REMOVE(str,template)

REPEAT
REPEAT(str,x)

REPLACE
REPLACE(str1,x,y,str2)

REVERSE
REVERSE(str)

RIGHT
RIGHT(str,x)

RINDEX
RINDEX(range,[sheet1 offset],col1 offset,row1 offset,[sheet2  ↩
    offset],col2 offset,row2 offset)

ROUND
ROUND(x,y)

ROW
ROW()  ROW(cell or range)  ROW(range,cell)

ROWS
ROWS(range)

SECOND
SECOND(time)

SETCOLOR
SETCOLOR(cell/range,color)
```

```
SETSTYLE
SETSTYLE(cell/range,style)

SHEET
SHEET()  SHEET(cell/range)  SHEET(range,cell)

SHEETS
SHEETS(range)

SIGN
SIGN(x)

SIN
SIN(x)

SINH
SINH(x)

SLN
SLN(cost,salvage price,life)

SQRT
SQRT(x)

STD
STD(multiple range/cell/expression arguments)

STDS
STDS(multiple range/cell/expression arguments)

STR
STR(number or string)

STRING
STRING(number,decimals)

SUM
SUM(multiple range/cell/expression arguments)

SYD
SYD(cost,salvage price,life,period)

SYMBOL
SYMBOL(str)

TAN
TAN(x)

TANH
TANH(x)

TERM
TERM(interest rate,future value,present value)

TERMA
TERMA(annuity,interest rate,future value)
```

```
THERE
THERE

TIME
TIME(hour,minute,second)

TIMESTR
TIMESTR(str,x)

TIMEVALUE
TIMEVALUE(str)

TODAY
TODAY

TRANSLATE
TRANSLATE(str,template1,template2)

TRIM
TRIM(str)

TRUE
TRUE

TSIGN
TSIGN(T value,degrees of freedom)

UPPER
UPPER(str)

VALUE
VALUE(str)

VAR
VAR(multiple range/cell/expression arguments)

VARS
VARS(multiple range/cell/expression arguments)

VLOOK
VLOOK(value to be looked up,range,column offset)

WINDCHILL
WINDCHILL(degrees Celcius,wind speed in kph)

WORD
WORD(str,n)

WORDS
WORDS(str)

YEAR
YEAR(date)
```

1.2 Formula Function: ABOVE...

```
                ABOVE
Value or address of cell above the current cell.

Format:
ABOVE

Input:
None

Output:
cell:
  address or value of cell above current cell

Example:
=above+1    = cell above + 1
=row(above) = row number of cell above

See Also:

                BELOW
                ,
                LEFT
                ,
                RIGHT
```

1.3 Formula Function: ABS...

```
ABS
Absolute Value of a number.

Format:
ABS(x)

Input:
number: x

Output:
number: absolute value of x

Example:
abs(-1) = 1
abs(1) = 1
```

1.4 Formula Function: ACOS...

```
ACOS
Arc cosine of a number.

Format:
ACOS(x)
```

```
Input:
number: x   (limits: -1.0 to 1.0)

Output:
number: arc cosine of x
ERROR:  if input exceed limits

Example:
acos(0.2) = 1.36943
```

1.5 Formula Function: ACOSH...

```
ACOSH
Inverse hyperbolic cosine of a number.

Format:
ACOSH(x)

Input:
number: x

Output:
number: inverse hyperbolic cosine of x

Example:
acosh(12) = 3.167313
```

1.6 Formula Function: ACOTH...

```
ACOTH
Inverse hyperbolic cotangent of x.

Format:
ACOSH(x)

Input:
number: x

Output:
number: inverse hyperbolic cotangent of x

Example:
acoth(12) = 0.083527042
```

1.7 Formula Function: ALPHA...

```
                ALPHA
Least Squares Method Function - Alpha Value.
```

```
Format:
ALPHA(range)

Input:
range:
  Two-column range, with a minimum of 2 rows.  The first
  column must contain the X values (independant variable),
  and the second column must contain the Y values
  (dependant variable)

Output:
number: 'a' value for the formula  y=a+bx

Example:
alpha(a1:b10)

See Also:

               BETA
               ,
               CORR
               ,
               COVAR
```

1.8 Formula Function: ASIN...

```
ASIN
Arc sine of a number.

Format:
ASIN(x)

Input:
number: x   (limits: -1.0 to 1.0)

Output:
number: arc sine of x
ERROR:  if input exceed limits

Example:
asin(0.2) = 0.201357921
```

1.9 Formula Function: ASINH...

```
ASINH
Inverse hyperbolic sine of a number.

Format:
ASINH(x)

Input:
number: x
```

```
Output:
number: inverse hyperbolic sine of x

Example:
asinh(12) = 3.179785438
```

1.10 Formula Function: ATAN...

```
ATAN
Arc tangent of a number.

Format:
ATAN(x)

Input:
number: x

Output:
number: Arc tangent of x

Example:
atan(12) = 1.487655095
```

1.11 Formula Function: ATAN2...

```
ATAN2
Arc tangent of two numbers.

Format:
ATAN2(x,y)

Input:
number: x
number: y

Output:
number: Arc tangent of x/y in the range -pi to +pi
ERROR:  If inputs x and y are both zero

Example:
atan2(1,2) = 0.463647609
```

1.12 Formula Function: ATANH...

```
ATANH
Inverse hyperbolic tangent of a number.

Format:
ATANH(x)
```

```
Input:
number: x (limits: x^2 <= 1.0)

Output:
number: inverse hyperbolic tangent of x
ERROR:  if x exceeds limits

Example:
atanh(0.2) = 0.202732554
```

1.13 Formula Function: AVG...

```
                AVG
Average of one or more numeric values.

Format:
AVG(multiple range/cell/expression arguments)

Input:
One or more of:
 - numbers
 - cells that contain numeric values
 - ranges (only cells with numeric contents in the range are used)

Null arguments, e.g. =avg(10,,5), are accepted, but are only
used to increase the 'count'.  The above formula will produce
5 (15/3) not 7.5 (15/2).

Output:
number:
  Average value of all arguments provided.  This is defined
  as the sum of all numeric values in the arguments provided
  divided by the 'count' of arguments provided.

  'count' is the number of numeric arguments, plus number
  of cells arguments, plus number of cells with contents
  in range arguments.

Example:
avg(a1:b10,12,100,a1/23))
avg(100,a1,b10:c20,cos(12))

See Also:

                COUNT
                ,
                SUM
                ,
                CAVG
                ,
                FAVG
```

1.14 Formula Function: BELOW...

```
                BELOW
Value or address of cell below the current cell.

Format:
BELOW

Input:
None

Output:
cell:
  address or value of cell above below cell

Example:
=below+1    = cell below + 1
=row(below) = row number of cell below

See Also:

                ABOVE
                ,
                LEFT
                ,
                RIGHT
```

1.15 Formula Function: BETA...

```
                BETA
Least Squares Method Function - Beta Value.

Format:
BETA(range)

Input:
range:
  Two-column range, with a minimum of 2 rows.  The first
  column must contain the X values (independant variable),
  and the second column must contain the Y values
  (dependant variable).

Output:
number: 'b' value for the formula  y=a+bx

Example:
beta(a1:b10)

See Also:

                ALPHA
                ,
                CORR
                ,
```

```
                COVAR
```

1.16 Formula Function: BIN...

```
                BIN
Convert a number to a binary string.

Format:
BIN(x)

Input:
number: x

Output:
string: binary string equivalent of x

Example:
bin(12) = "1100"

See Also:

                BINTO
```

1.17 Formula Function: BINTO...

```
                BINTO
Convert a binary string to a number.

Format:
BINTO(str)

Input:
string: str (consisting of 1's and 0's)

Output:
number: value of binary number in str

Example:
binto("1100") = 12

See Also:

                BIN
```

1.18 Formula Function: CASE...

```
                CASE
Select an entry from a list.

Format:
```

```
CASE(x,v1,v2,v3,....,vn)

Input:
number:
  x   (limits: x >= 1)
entry:
  one or more numbers and/or strings
  v1, v2, v3, ... vn

Output:
number:
  returns v1 if x = 1, v2 if x = 2, v3 if x = 3, etc.
ERROR:
  if x < 1 or x > number of values

CASE() will also return a cell or range address if used as a
cell or range argument to another function.

Example:
case(2,11,22,33,44) = 22
case(1,11,22,33,44) = 11
case(3,"one","two","three","four") = "three"

See Also:

                PICK
                ,
                CHOOSE
```

1.19 Formula Function: CAVG...

```
                CAVG
Average of multiple numbers conditionally selected.

Format:
CAVG(expression,multiple range arguments)

Input:
expression:
  a formula expression.  This is tested for every
  cell in 'arguments' below, and if the expression is
  true (non-zero) then the cell is considered as a
  valid argument, otherwise the cell is ignored.

arguments:
  one or more data ranges

Output:
number:
  average value of all numeric cells for which
  expression was true.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:
```

```
=cavg(here>5,a1:a10)

This will give you the average of all cells in the range a1:a10
which have a value higher than 5.

=cavg((here>=5)&(here<=10),a1:a10)

This will give you the average of cells in the range a1:a10
which have a value between 5 and 10.

See Also:

                AVG
                ,
                CCOUNT
                ,
                CSUM
                ,
                FAVG
```

1.20 Formula Function: CCOUNT...

```
                CCOUNT
Count number of cells within a range that meet specific criteria.

Format:
CCOUNT(expression,multiple range arguments)

Input:
expression:
  a formula expression.  This is tested for every
  cell in 'arguments' below, and if the expression is
  true (non-zero) then the cell is considered as a
  valid argument, otherwise the cell is ignored.

arguments:
  one or more data ranges

Output:
number:
  number of valid numeric cells for which expression
  was true.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

=ccount(here>5,a1:a10)

This will give you the number of cells in the range a1:a10
which have a value higher than 5.

=ccount((here>=5)&(here<=10),a1:a10)
```

```
This will give you the number of cells in the range a1:a10
which have a value between 5 and 10.

See Also:

                COUNT
                ,
                CAVG
                ,
                CSUM
```

1.21 Formula Function: CELL...

```
                CELL
Get value of a specific cell in current sheet or another sheet.

Format:
CELL(column,row)
CELL(sheet,column,row)

Input:
sheet:  [optional] sheet number from 1 to 254
column: column number, from 0 to 18277
row:  row number, from 1 to 65000

Output:
contents of the cell defined by arguments column and row in
the current sheet, or in another sheet (if 'sheet' is used).

Example:
cell(0,1) = value of cell a1 in the current sheet
cell(2,0,1) = value of cell a1 in sheet B

Notes:
CELL() is a LIVE function.  It is evaluated at every recalc.

CELL() does not guarantee the returned value is totally correct
if Minimal Recalc is used.  It is guaranteed to return the
correct value only if Full Recalc is used.

The INDEX() function is generally much safer.

See Also:

                INDEX
```

1.22 Formula Function: CFIRST...

```
                CFIRST
Get first cell within a range that meets specific criteria.

Format:
```

```
CFIRST(expression,multiple range arguments)

Input:
expression:
  a formula expression.  This is tested for every
  cell in 'arguments' below, and if the expression is
  true (non-zero) then the cell is considered as a
  valid argument, otherwise the cell is ignored.

arguments:
  one or more data ranges

Output:
number or string:
  value of the first cell for which expression was true.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

=cfirst(here>5,a1:a10)

This will give you the first cell in the range a1:a10 that has
a value higher than 5.

=cfirst((here>=5)&(here<=10),a1:a10)

This will give you the value of the first cell in the range
a1:a10 that has a value between 5 and 10.

See Also:

                FIRST
                ,
                CLAST
                ,
                CPICK
```

1.23 Formula Function: CHAR...

```
                CHAR
Get ASCII character.

Format:
CHAR(x)

Input:
number: x (x between 0 and 255)

Output:
string: single character string representing ASCII value of x.

Example:
char(75) = "K"
```

```
See Also:

                CODE
```

1.24 Formula Function: CHOOSE...

```
                CHOOSE
Select an entry from a list.

Format:
CHOOSE(x,v1,v2,v3,....,vn)

Input:
number:
  x   (limits: x >= 1)
entry:
  one or more numbers and/or strings
  v1, v2, v3, ... vn

Output:
number: returns v1 if x = 1, v2 if x = 2, v3 if x = 3, etc.
ERROR:  if x < 1 or x > number of values

CHOOSE() will also return a cell or range address if used as a
cell or range argument to another function.

Example:
choose(2,11,22,33,44) = 22
choose(1,11,22,33,44) = 11
choose(3,"one","two","three","four") = "three"

See Also:

                PICK
                ,
                CASE
```

1.25 Formula Function: CLAST...

```
                CLAST
Get last cell within a range that meets specific criteria.

Format:
CLAST(expression,multiple range arguments)

Input:
expression:
  a formula expression.  This is tested for every
  cell in 'arguments' below, and if the expression is
  true (non-zero) then the cell is considered as a
  valid argument, otherwise the cell is ignored.
```

```
arguments:
  one or more data ranges

Output:
number or string:
  value of the last cell for which expression was true.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

=clast(here>5,a1:a10)

This will give you the last cell in the range a1:a10 that has
a value higher than 5.

=clast((here>=5)&(here<=10),a1:a10)

This will give you the value of the last cell in the range
a1:a10 that has a value between 5 and 10.

See Also:

               LAST
               ,
               CFIRST
               ,
               CPICK
```

1.26 Formula Function: CLEAN...

```
CLEAN
Clean a string from any non-printable characters.

Format:
CLEAN(str)

Input:
string: str

Output:
string:
  identical to 'str' except that any non-printable
  characters are removed.  Non-printable characters are
  defined as ASCII characters 0 to 31 and 128 to 159.
```

1.27 Formula Function: CMAX...

```
               CMAX
Highest value of multiple numbers conditionally selected.

Format:
```

```
CMAX(expression,multiple range arguments)

Input:
expression:
  a formula expression.  This is tested for every
  cell in 'arguments' below, and if the expression is
  true (non-zero) then the cell is considered as a
  valid argument, otherwise the cell is ignored.

arguments:
  one or more data ranges

Output:
number:
  highest value of all numeric cells for which
  expression was true.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

=cmax(here<5,a1:a10)

This will give you the highest value of all cells in the range
a1:a10 which have a value lower than 5.

=cmax((here>=5)&(here<=10),a1:a10)

This will give you the highest value of cells in the range
a1:a10 which have a value between 5 and 10.

See Also:

                MAX
                ,
                CMIN
```

1.28 Formula Function: CMIN...

```
                CMIN
Lowest value of multiple numbers conditionally selected.

Format:
CMIN(expression,multiple range arguments)

Input:
expression:
  a formula expression.  This is tested for every
  cell in 'arguments' below, and if the expression is
  true (non-zero) then the cell is considered as a
  valid argument, otherwise the cell is ignored.

arguments:
  one or more data ranges
```

```
Output:
number:
  lowest value of all numeric cells for which expression
  was true.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

=cmin(here>5,a1:a10)

This will give you the lowest value of all cells in the range
a1:a10 which have a value higher than 5.

=cmin((here>=5)&(here<=10),a1:a10)

This will give you the lowest value of cells in the range
a1:a10 which have a value between 5 and 10.

See Also:

                MIN
                ,
                CMAX
```

1.29 Formula Function: CODE...

```
                CODE
Get ASCII code of a character.

Format:
CODE(str)

Input:
string: str (only first character in str is used)

Output:
number: ASCII value of first character in str

Example:
code("t") = 116

See Also:

                CHAR
```

1.30 Formula Function: COL...

```
                COL
Get column number of a cell or range.

Format:
```

```
COL()
COL(cell)
COL(range)
COL(range,cell)

Input:
cell: [optional] a cell name
range:  [optional] a range name

Output:
Depends on arguments provided:

COL()
returns the column number of the current cell.

COL(cell)
returns the column number of the provided cell.

COL(range)
returns the column number of the top left cell in the provided range.

COL(range,cell)
returns the column offset of 'cell' in 'range', or ERROR if
'cell' is outside 'range'.

Example:
col() = column number of current cell
col(c1) = 3 (column C)
col(c1:d1) = 3 (column C)
col(c1:e10,d1) = 1 (offset of column D to column C)

See Also:

                COLS
                ,
                ROW
```

1.31 Formula Function: COLOR...

```
                COLOR
Get the Graph Color number.

Format:
COLOR(str)

Input:
string:
  str (string matching the name of a Graph Color)

Output:
number:
  number of color in graph colors list, or 0 if it was
  not found.

Example:
```

```
color("yellow") = 12 (number of color 'yellow')

color(if(a1>=0,"yellow","light red"))
  this will return the value of 'yellow' (12) if a1>=0,
  otherwise it will return the value of 'light red' (2).

See Also:

                PATTERN
                ,
                SYMBOL
```

1.32 Formula Function: COLS...

```
                COLS
Get the number of columns in a range.

Format:
COLS(range)

Input:
range:  a range name

Output:
number: number of columns in range

Example:
cols(a1:c1) = 3

See Also:

                ROWS
                ,
                SHEETS
```

1.33 Formula Function: COMB...

```
                COMB
Calculate the Combination possibilities of a selection of
distinct items from a group.

Format:
COMB(n,r)

Input:
number: n
number: r

Output:
number:
  the Combination value (as used in Probability and
  Statistics) which is usually defined as  nCr = nPr/r!
```

```
  (where nPr = n!/(n-r)!)

Example:
If you have a group of five people: A, B, C, D and E and
wanted to select only three of them.  There would be
comb(5,3) = 12 possible selections:
ABC ABD ABE ACD ACE ADE BCD BCE BDE and CDE

Note:
Rearranging items within an Combination does not give a
different combination, thus ABC and ACB are the same
combination.

See Also:

                PERM
```

1.34 Formula Function: CORR...

```
                CORR
Least Squares Method Function - Correlation Co-effecient.

Format:
CORR(range)

Input:
range:
  Two-column range, with a minimum of 2 rows.  The first
  column must contain the X values (independant variable),
  and the second column must contain the Y values
  (dependant variable)

Output:
number:
  Correlation Co-effecient using the Least Squares Method
  between the dependant and independant variables.

Example:
corr(a1:b10)

See Also:

                ALPHA
                ,
                BETA
                ,
                COVAR
```

1.35 Formula Function: COS...

```
COS
Cosine of a number.
```

```
Format:
COS(x)

Input:
number: x

Output:
number: cosine of x

Example:
cos(12) = 0.843853959
```

1.36 Formula Function: COSH...

```
COSH
Hyperbolic cosine of a number.

Format:
COSH(x)

Input:
number: x (limit: x <= 709.778)

Output:
number: hyperbolic cosine of x
ERROR:  if x exceeds limits

Example:
cosh(12) = 81377.3957
```

1.37 Formula Function: COTAN...

```
COTAN
Cotangent of a number.

Format:
COTAN(x)

Input:
number: x

Output:
number: cotangent of x

Example:
cotan(12) = -1.572673406
```

1.38 Formula Function: COTH...

```
COTH
Hyperbolic cotangent of a number.

Format:
COTH(x)

Input:
number: x

Output:
number: hyperbolic cotangent of x

Example:
coth(1) = 1.313035285
```

1.39 Formula Function: COUNT...

```
                COUNT
Count of one or more numeric values.

Format:
COUNT(multiple range/cell/expression arguments)

Input:
One or more of:
 - numbers
 - cells that contain numeric values
 - ranges (only cells with numeric contents in the range are used)

Null arguments, e.g. =count(10,,5), are accepted, but are only
used to increase the 'count'.  The above formula will produce
a count of 3, not 2.

Output:
number:
  Number of valid numeric arguments provided.  This is
  defined as the number of numeric arguments, plus number
  of cells arguments, plus number of cells with contents
  in range arguments.

Example:
count(a1:b10,12,100)
count(100,a1,b10:c20)

See Also:

                AVG
                ,
                SUM
                ,
                CCOUNT
                ,
                FCOUNT
```

1.40 Formula Function: COVAR...

```
                COVAR
Least Squares Method Function - Co-variance.

Format:
COVAR(range)

Input:
range:
  Two-column range, with a minimum of 2 rows.  The first
  column must contain the X values (independant variable),
  and the second column must contain the Y values
  (dependant variable)

Output:
number:
  Covariance using the Least Squares Method between the
  dependant and independant variables.

Example:
covar(a1:b10)

See Also:

                ALPHA
                ,
                BETA
                ,
                CORR
```

1.41 Formula Function: CPICK...

```
                CPICK
Get a specific cell within a range that meets specific criteria.

Format:
CPICK(offset,expression,multiple range arguments)

Input:
number:
  offset, a positive or negative integer number.
  It is used to define which cell to select from
  the cells for which expression is true.

  If offset is positive, it picks cells from the
  start of the ranges defined.  If it is negative,
  it picks cells from the end of the ranges defined.

    3 use the 3rd argument
    2 use the 2nd argument
    1 use the 1st argument
    0 use the last argument
   -1 use the argument before last
```

```
   -2 use the 2nd argument before last
   -3 use the 3rd argument before last

expression:
  a formula expression.  This is tested for every
  cell in 'arguments' below, and if the expression is
  true (non-zero) then the cell is considered as a
  valid argument, otherwise the cell is ignored.

arguments:
  one or more data ranges

Output:
number or string:
  value of the selected cell for which expression was
  true.  If no cell was found, 0 is returned.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

=cpick(3,here>5,a1:a10)

This will give you the value of the third cell in the range
a1:a10 which has a value higher than 5.

=cpick(-2,(here>=5)&(here<=10),a1:a10)

This will give you the value of the 2nd to last cell in the
range a1:a10 which has a value between 5 and 10.

See Also:

                PICK
                ,
                CFIRST
                ,
                CLAST
```

1.42 Formula Function: CSUM...

```
                CSUM
Sum of multiple numbers conditionally selected.

Format:
CSUM(expression,multiple range arguments)

Input:
expression:
  a formula expression.  This is tested for every
  cell in 'arguments' below, and if the expression is
  true (non-zero) then the cell is considered as a
  valid argument, otherwise the cell is ignored.

arguments:
```

```
  one or more data ranges

Output:
number:
  sum of all numeric cells for which expression was true.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

=csum(here>5,a1:a10)

This will give you the sum of all cells in the range a1:a10
which have a value higher than 5.

=csum((here>=5)&(here<=10),a1:a10)

This will give you the sum of cells in the range a1:a10
which have a value between 5 and 10.

See Also:

                SUM
                ,
                CAVG
                ,
                CCOUNT
                ,
                FSUM
```

1.43 Formula Function: DATE...

```
                DATE
Calculate the serial date number for a specific year, month and
day.

Format:
DATE(year,month,day)

Input:
number: year (from 0 to 2100)
number: month (from 1 to 12)
number: day (from 1 to 31)

Output:
date:
  serial date number for the day, month and year selected.

Example:
date(1994,2,1) = 5875, which is the serial date number for
1-feb-1994.

See Also:

                YEAR
```

,
MONTH
,
DAY
,
TODAY
,
DATESTR
,
DATEVALUE

1.44 Formula Function: DATESTR...

```
                DATESTR
Convert a date number to a string using a specific format.

Format:
DATESTR(str,date)

Input:
string: str (string containing a date format specification)
date: date (serial date number)

Output:
string:
  a string describing the date provided in the format
  provided.

Example:
datestr("dd Mmm yyyy",TODAY) would return "01 Feb 1994"

See Also:

                DATE
                ,
                DATEVALUE
```

1.45 Formula Function: DATEVALUE...

```
                DATEVALUE
Convert date string to a date serial number.

Format:
DATEVALUE(date string)

Input:
string: date string
  The date string should look like a date, i.e. day
  number, month name (3 characters at least) and year
  number.  Indirect date specifications are also
  accepted,  e.g. "next monday".  See Appendix 6 for
  details on date strings.
```

```
Output:
date: date serial number
ERROR:  if date string was not valid

Example:
datevalue("1 feb 1994") =  5875, which is the serial date
number for 1-feb-1994.

Indirect date specification:

datevalue("next monday")
datevalue("second monday in jan 94")
datevalue("friday")
datevalue("next monday")
datevalue("last monday")
datevalue("second last monday")
datevalue("today")
datevalue("first monday in jan 94")
datevalue("last monday in feb")
datevalue("first tuesday next month")
datevalue("last tuesday this month")
datevalue("third friday last month")
datevalue("first day next year")

See Also:

               DATESTR
               ,
               DATE
```

1.46 Formula Function: DAVG...

```
               DAVG
Average value of a selected field in a database.

Format:
DAVG(offset)
DAVG(database input range, offset)
DAVG(offset, database criteria range)
DAVG(database input range, offset, database criteria range)

Input:
range: [optional]
  Database input range.  If this is not defined, the default Data Input Range
  for the current sheet is used.

number:
  offset, selects the field (column) to use from the database input range.

range: [optional]
  Database critiera range.  If this is not defined, the default Data Criteria
  Range for the current sheet is used.

Output:
```

```
number:
  average value of all entries in the selected field
  (column) in the database entries that match the
  database Search Criteria.
ERROR:
  if the database input or criteria ranges are not defined.

See Also:

                DCOUNT
                ,
                DSUM
                ,
                DMIN
                ,
                DMAX
                ,
                DVAR
                ,
                DSUM
```

1.47 Formula Function: DAY...

```
                DAY
Get day number from a serial date number.

Format:
DAY(date)

Input:
date: serial date number

Output:
number:
  from 1 to 31, the day number from the provided serial
  date number

Example:
day(today) = 12  (if today was the 12th of the month)

See Also:

                DATE
                ,
                MONTH
                ,
                YEAR
                ,
                TODAY
                ,
                DATESTR
                ,
                DATEVALUE
```

1.48 Formula Function: DAYS...

```
                DAYS
Get the number of days in the month specified.

Format:
DAYS(year,month)

Input:
number: year
number: month

Output:
number:
  number of days in the month and year specified.
  This takes into consideration leap years.

Example:
days(1994,1) = 31 (days in the month of Jan 1994)

To get the number of days in the current month:
days(year(today),month(today))

See Also:

                DATE
                ,
                DAY
                ,
                MONTH
                ,
                YEAR
```

1.49 Formula Function: DCOUNT...

```
                DCOUNT
Number of entries that match the search criteria in a selected
field in a database.

Format:
DCOUNT(offset)
DCOUNT(database input range, offset)
DCOUNT(offset, database criteria range)
DCOUNT(database input range, offset, database criteria range)

Input:
range: [optional]
  Database input range.  If this is not defined, the default Data Input Range
  for the current sheet is used.

number:
  offset, selects the field (column) to use from the database input range.

range: [optional]
```

```
  Database critiera range.  If this is not defined, the default Data Criteria
  Range for the current sheet is used.

Output:
number:
  number of entries in the selected field (column) in the
  database entries that match the database Search
  Criteria.
ERROR:
  if the database input or criteria ranges are not defined.

See Also:

                DSUM
                ,
                DMIN
                ,
                DMAX
                ,
                DAVG
```

1.50 Formula Function: DEC...

```
                DEC
Convert decimal number to a string.

Format:
DEC(x)

Input:
number: x

Output:
string: a string with the decimal value of x

Example:
dec(123) = "123"

See Also:

                STRING
                ,
                VALUE
                ,
                HEX
                ,
                BIN
```

1.51 Formula Function: DEGREES...

```
                DEGREES
Convert a number from radians to degrees.
```

```
Format:
DEGREES(x)

Input:
number: x

Output:
number: value of x radians in degrees

Example:
degrees(12) = 114.592

See Also:

                RADIANS
```

1.52 Formula Function: DMAX...

```
                DMAX
Highest value of a selected field in a database.

Format:
DMAX(offset)
DMAX(database input range, offset)
DMAX(offset, database criteria range)
DMAX(database input range, offset, database criteria range)

Input:
range: [optional]
  Database input range.  If this is not defined, the default Data Input Range
  for the current sheet is used.

number:
  offset, selects the field (column) to use from the database input range.

range: [optional]
  Database critiera range.  If this is not defined, the default Data Criteria
  Range for the current sheet is used.

Output:
number:
  highest value of all entries in the selected field
  (column) in the database entries that match the
  database Search Criteria.
ERROR:
  if the database input or criteria ranges are not defined.

See Also:

                DCOUNT
                ,
                DMIN
                ,
                DAVG
```

,
DVAR
,
DSUM

1.53 Formula Function: DMIN...

```
                DMIN
Lowest value of a selected field in a database.

Format:
DMIN(offset)
DMAX(database input range, offset)
DMAX(offset, database criteria range)
DMAX(database input range, offset, database criteria range)

Input:
range: [optional]
  Database input range.  If this is not defined, the default Data Input Range
  for the current sheet is used.

number:
  offset, selects the field (column) to use from the database input range.

range: [optional]
  Database critiera range.  If this is not defined, the default Data Criteria
  Range for the current sheet is used.


Output:
number:
  lowest value of all entries in the selected field
  (column) in the database entries that match the
  database Search Criteria.
ERROR:
  if the database input or criteria ranges are not defined.

See Also:

                DCOUNT
                ,
                DMAX
                ,
                DAVG
                ,
                DVAR
                ,
                DSUM
```

1.54 Formula Function: DSTD...

```
                DSTD
Standard deviation of a selected field in a database.

Format:
DSTD(offset)
DSTD(database input range, offset)
DSTD(offset, database criteria range)
DSTD(database input range, offset, database criteria range)

Input:
range: [optional]
  Database input range.  If this is not defined, the default Data Input Range
  for the current sheet is used.

number:
  offset, selects the field (column) to use from the database input range.

range: [optional]
  Database critiera range.  If this is not defined, the default Data Criteria
  Range for the current sheet is used.


Output:
number:
  standard deviation of all entries in the selected field
  (column) in the database entries that match the database
  Search Criteria.
ERROR:
  if the database input or criteria ranges are not defined.

See Also:

                DCOUNT
                ,
                DMAX
                ,
                DMIN
                ,
                DAVG
                ,
                DVAR
```

1.55 Formula Function: DSUM...

```
                DSUM
Sum of a selected field in a database.

Format:
DSUM(offset)
DSUM(database input range, offset)
DSUM(offset, database criteria range)
DSUM(database input range, offset, database criteria range)

Input:
```

```
range: [optional]
  Database input range.  If this is not defined, the default Data Input Range
  for the current sheet is used.

number:
  offset, selects the field (column) to use from the database input range.

range: [optional]
  Database critiera range.  If this is not defined, the default Data Criteria
  Range for the current sheet is used.


Output:
number:
  sum of all entries in the selected field (column) in
  the database entries that match the database Search
  Criteria.
ERROR:
  if the database input or criteria ranges are not defined.

See Also:

                DCOUNT
                ,
                DMAX
                ,
                DMIN
                ,
                DAVG
                ,
                DVAR
```

1.56 Formula Function: DVAR...

```
                DVAR
Variance of a selected field in a database.

Format:
DVAR(offset)
DVAR(database input range, offset)
DVAR(offset, database criteria range)
DVAR(database input range, offset, database criteria range)

Input:
range: [optional]
  Database input range.  If this is not defined, the default Data Input Range
  for the current sheet is used.

number:
  offset, selects the field (column) to use from the database input range.

range: [optional]
  Database critiera range.  If this is not defined, the default Data Criteria
  Range for the current sheet is used.
```

```
Output:
number:
  variance of all entries in the selected field (column) in
  the database entries that match the database Search
  Criteria.
ERROR:
  if the database input or criteria ranges are not defined.

See Also:

                DCOUNT
                ,
                DMAX
                ,
                DMIN
                ,
                DAVG
                ,
                DSUM
                ,
                DSTD
```

1.57 Formula Function: ERROR...

```
                ERROR
Set ERROR condition in a formula.

Format:
ERROR

Input:
None

Output:
ERROR:  set result to ERROR

Example:
if(a1<10,a1,ERROR)  this will return the value of a1 if a1
is less than 10, or ERROR if it isn't.

See Also:

                IF
                ,
                ISERR
                ,
                IZERR
```

1.58 Formula Function: EVALUATE...

```
                EVALUATE
Evaluate a string as a formula.

Format:
EVALUATE(expression)

Input:
string:
  expression: any valid formula or math expression
  You can use any cell or range reference within
  the expression.

Output:
result:
  number or string depending on result of evaluated
  expression
ERROR:
  if expression was not valid

Example:
evaluate("1+2+(12^7)")

Note:
EVALUATE() is a live function, it is evaluated at every recalc.

See Also:

                VALUE
```

1.59 Formula Function: EXACT...

```
EXACT
Compare two strings (case-sensitive).

Format:
EXACT(str1,str2)

Input:
string: str1
string: str2

Output:
TRUE:
  if str1 and str2 are identical
FALSE:
  if str1 and str2 are not identical

Example:
EXACT() does a case-sensitive comparison, unlike the normal
string comparison using the '=' sign.

"test"="Test"     returns TRUE
exact("test","Test")  returns FALSE
```

1.60 Formula Function: EXP...

```
EXP
Value of e (2.718281828) to the power x.

Format:
EXP(x)

Input:
number: x (limits: x between -708.396 and 709.778)

Output:
number: value of e (2.718281828) to the power x
ERROR:  if x exceeds limits

Example:
exp(12) = 162754.791
```

1.61 Formula Function: FACT...

```
FACT
Calculate Factorial of a number.

Format:
FACT(x)

Input:
number:
  x (limits: x >= 0.0)
  any fraction is removed before x is used, i.e.
  fact(2.1) = fact(2)

Output:
number:
  factorial of x
1:
  if x = 0
ERROR:
  if x < 0

Example:
fact(3)   equals 1 x 2 x 3, i.e. 6
fact(4)   equals 1 x 2 x 3 x 4, i.e. 24
fact(12)  equals 1 x 2 x 3 x ... x 12, i.e. 479001600
```

1.62 Formula Function: FALSE...

```
                FALSE
Constant: value of FALSE (zero).

Format:
FALSE
```

```
Input:
None

Output:
FALSE:  (zero)

Example:
if(a1<12,TRUE,FALSE)

See Also:

                TRUE
```

1.63 Formula Function: FAVG...

```
                FAVG
Average of applying an expression to one or more arguments.

Format:
FAVG(expression,multiple range arguments)

Input:
expression:
  a formula expression.  This is evaluated for every
  cell in 'arguments' below, and the result is used
  as the argument in the calculations.

arguments:
  one or more data ranges

Output:
number:
  average value of the results of evaluating the
  expression for all numeric cells in the arguments
  provided.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

favg(here*2,a1:a10)

This will give you the average of all cells in the range a1:a10
multiplied by 2.

favg(cos(here),a1:a10)

This will give you the average of cosines of the cells in the
range a1:a10.

See Also:

                AVG
                ,
```

CAVG
,
FCOUNT
,
FSUM

1.64 Formula Function: FCOUNT...

FCOUNT
Number of numeric cells in one or more ranges.

Format:
FAVG(expression,multiple range arguments)

Input:
expression:
  ignored by FCOUNT()
arguments:
  one or more data ranges

Output:
number:
  average value of the results of evaluating the
  expression for all numeric cells in the arguments
  provided.

Note:
This is a dummy function. It works exactly the same as
COUNT(). 'expression' is ignored. It is only provided as a
way to easily debug the FAVG(), FSUM(), FMAX(), FMIN(), etc.
functions. It is often useful to quickly change FSUM() to
FCOUNT() to see how many entries are affecting it.

See Also:

FAVG
,
FSUM
,
FSUM
,
COUNT

1.65 Formula Function: FFIRST...

FFIRST
Result of applying an expression to the first cell in one or
more arguments.

Format:
FFIRST(expression,multiple range arguments)

```
Input:
expression:
  a formula expression.  This is evaluated for every
  cell in 'arguments' below, and the result is used
  as the argument in the calculations.

arguments:
  one or more data ranges

Output:
number:
  result of evaluating the expression for the first
  valid cell in the arguments provided.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

ffirst(here*2,a1:a10)

This will give you the value of the first valid cell in the
range a1:a10 multiplied by 2.

ffirst(cos(here),a1:a10)

This will give you the cosine of the first valid cell in the
range a1:a10.

See Also:

               FLAST
               ,
               FPICK
               ,
               FIRST
               ,
               CFIRST
```

1.66 Formula Function: FIELD...

```
FIELD
Value of current field being evaluated in the current database.

Format:
FIELD

Input:
None

Output:
Value of current field being evaulated.  Used in Database
operations only.

Example:
```

See the database section for a detailed description.

1.67 Formula Function: FIND...

```
FIND
Find a string in another string.

Format:
FIND(str1,str2,n)

Input:
string: str1
string: str2
number: n

Output:
number:
  starting position of first occurance of 'str2' in
  'str1' starting from the n'th character
ERROR:
  if str2 was not found in str1

Example:
find("this is a test","test",0) returns 10
```

1.68 Formula Function: FIRST...

```
                FIRST
Get first valid entry from multiple numeric or string arguments.

Format:
FIRST(multiple range/cell/expression arguments)

Input:
One or more of:
 - numbers
 - cells
 - ranges

Output:
number or string: value of first valid entry from arguments

Example:
first(a1:b10) returns the value of the first cell in the range
a1:b10 that contains a valid value

See Also:

                LAST
                ,
                PICK
                ,
```

```
                FFIRST
                ,
                CFIRST
```

1.69 Formula Function: FLAST...

```
                FLAST
Result of applying an expression to the last cell in one or
more arguments.

Format:
FLAST(expression,multiple range arguments)

Input:
expression:
  a formula expression.  This is evaluated for every
  cell in 'arguments' below, and the result is used
  as the argument in the calculations.

arguments:
  one or more data ranges

Output:
number:
  result of evaluating the expression for the last
  valid cell in the arguments provided.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

flast(here*2,a1:a10)

This will give you the value of the last valid cell in the
range a1:a10 multiplied by 2.

flast(cos(here),a1:a10)

This will give you the cosine of the last valid cell in the
range a1:a10.

See Also:

                FFIRST
                ,
                FPICK
                ,
                LAST
                ,
                CLAST
```

1.70 Formula Function: FMAX...

```
                FMAX
Largest result of applying an expression to the first cell in
one or more arguments.

Format:
FMAX(expression,multiple range arguments)

Input:
expression:
  a formula expression.  This is evaluated for every
  cell in 'arguments' below, and the result is used
  as the argument in the calculations.

arguments:
  one or more data ranges

Output:
number:
  largest result of evaluating the expression to all
  valid cells in the arguments provided.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

fmax(here*2,a1:a10)

This will give you the value of the highest cell value in the
range a1:a10 multiplied by 2.

fmax(cos(here),a1:a10)

This will give you the highest cosine value of all cells in the
range a1:a10.

See Also:

                FMIN
                ,
                MAX
                ,
                CMAX
```

1.71 Formula Function: FMIN...

```
                FMIN
Lowest result of applying an expression to the first cell in
one or more arguments.

Format:
FMIN(expression,multiple range arguments)

Input:
expression:
```

```
  a formula expression.  This is evaluated for every
  cell in 'arguments' below, and the result is used
  as the argument in the calculations.

arguments:
  one or more data ranges

Output:
number:
  lowest result of evaluating the expression to all
  valid cells in the arguments provided.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

fmin(here*2,a1:a10)

This will give you the value of the lowest cell value in the
range a1:a10 multiplied by 2.

fmin(cos(here),a1:a10)

This will give you the lowest cosine value of all cells in the
range a1:a10.

See Also:

                FMAX
                ,
                MIN
                ,
                CMIN
```

1.72 Formula Function: FPICK...

```
                FPICK
Result of applying an expression to the a specific cell in one
or more arguments.

Format:
FPICK(offset,expression,multiple range arguments)

Input:
number:
  offset, a positive or negative integer number.
  It is used to define which cell to select from
  the cells for which expression is true.

  If offset is positive, it picks cells from the
  start of the ranges defined.  If it is negative,
  it picks cells from the end of the ranges defined.

    3 use the 3rd argument
    2 use the 2nd argument
```

```
    1 use the 1st argument
    0 use the last argument
   -1 use the argument before last
   -2 use the 2nd argument before last
   -3 use the 3rd argument before last

expression:
  a formula expression.  This is evaluated for every
  cell in 'arguments' below, and the result is used
  as the argument in the calculations.

arguments:
  one or more data ranges

Output:
number:
  result of evaluating the expression for the
  selected cell in the arguments provided.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

fpick(2,here*2,a1:a10)

This will give you the value of the second valid cell in the
range a1:a10 multiplied by 2.

flast(-3,cos(here),a1:a10)

This will give you the cosine of the third last valid cell in
the range a1:a10.

See Also:

                FFIRST
                ,
                FLAST
                ,
                PICK
                ,
                CPICK
```

1.73 Formula Function: FRAC...

```
                FRAC
Get fractional part of a number.

Format:
FRAC(x)

Input:
number: x

Output:
```

```
number: fractional part of x

Example:
frac(1.45) = 0.45

See Also:

                INT
```

1.74 Formula Function: FSIGN...

```
                FSIGN
Get Significance of F Value given two degrees of freedom.
(used for Multivariate Regression Analysis)

Format:
FSIGN(f value,degrees of freedom 1,degrees of freedom 2)

Input:
number: f value
number: degrees of freedom 1
number: degrees of freedom 2

Output:
number:
  Significance of F value given the provided degrees of
  freedom.

See Also:

                REGRES
                ,
                TSIGN
```

1.75 Formula Function: FSUM...

```
                FSUM
Sum of applying an expression to one or more arguments.

Format:
FSUM(expression,multiple range arguments)

Input:
expression:
  a formula expression.  This is evaluated for every
  cell in 'arguments' below, and the result is used
  as the argument in the calculations.

arguments:
  one or more data ranges

Output:
```

```
number:
  sum of the results of evaluating the expression for
  all numeric cells in the arguments provided.

Example:
You can use the HERE keyword to mean the current cell being
evaluated.  For example:

fsum(here*2,a1:a10)

This will give you the sum of all cells in the range a1:a10
multiplied by 2.

fsum(cos(here),a1:a10)

This will give you the sum of cosines of the cells in the
range a1:a10.

See Also:

                SUM
                ,
                CSUM
                ,
                FAVG
                ,
                FCOUNT
```

1.76 Formula Function: FV...

```
                FV
Calculates the Future Value of an amount.

Format:
FV(amount,interest rate,number of periods)

Input:
number: amount
number: interest rate
number: periods

Output:
number:
  Future Value of a future amount with a specific
  number of periods and a specific interest rate using
  compound interest.

Example:
An amount of $10,000 today, would be worth $23,965.58 in 15
years at an interest rate of 6%.  (with interest compounded
yearly)

fv(10000,6%,15) = 23965.58

If interest is compounded monthly, then divide the interest
```

rate by 12, and multiply the periods by 12.

See Also:

FVA
,
PV
,
TERM
,
RATE

1.77 Formula Function: FVA...

FVA
Calculates the Future Value of a stream of fixed payments.

Format:
FVA(annuity amount,interest rate,number of periods)

Input:
number: annuity amount
number: interest rate
number: periods

Output:
number:
  Future Value of a stream of fixed payments over a
  number of periods presuming a specific interest rate
  using compound interest.

Example:
An annuity of $1,000 a month over 15 years at an interest rate
of 6% is worth $290,818.71 after 15 years.

pva(1000,6%/12,15*12) = 290818.71

See Also:

FVA
,
PV
,
TERMA
,
RATEA

1.78 Formula Function: GAMMA...

GAMMA
Calculate Gamma value of a number.

```
Format:
GAMMA(x)

Input:
number:
  x (limits: x >= 0.0)

Output:
number:
  Gamma value of x
  This is defined as the product of multiplying all
  numbers from 1 to x-1.
1:
  if x = 0
ERROR:
  if x < 0

Example:
gamma(6) = 1 x 2 x 3 x 4 x 5 = 120
```

1.79 Formula Function: GETCOLOR...

```
GETCOLOR
Get color of a cell.

Format:
GETCOLOR(cell)

Input:
cell: cell name

Output:
number:
  value that defines cell style:
    -1  cell not allocated
     0  default
     1  white
     2  red
     3  green
     4  blue
     5  orange
     6  pink
     7  cyan
     8  yellow

Example:
getcolor(a1) = 2 (if the color of cell a1 is red)

Note:
Since changes to cell color do not invoke a recalc, you
will need to issue a recalc command if you manually change
a cell's color.
```

1.80 Formula Function: GETSTYLE...

```
GETSTYLE
Get style of a cell.

Format:
GETSTYLE(cell)

Input:
cell: cell name

Output:
number:
  value that defines cell style:
    -1  cell not allocated
     0  normal
     1  bold
     2  italic
     4  undeline
     8  reverse
    16  shaded

  Combined styles are added together:
    bold+italic  = 1+2  = 3

Example:
getstyle(a1) = 1 (if the cell a1 is set to bold)

Note:
Since changes to cell style do not invoke a recalc, you
will need to issue a recalc command if you manually change
a cell's style.
```

1.81 Formula Function: HERE...

```
HERE
Address/value of current cell.

Format:
HERE
HERE(column offset,row offset)
HERE(sheet offset,column offset,row offset)

Input:
number: [optional] sheet offset
number: [optional] column offset
number: [optional] row offset

Output:
contents of the current cell in the current sheet, plus the
sheet, column and row offsets provided.

Example:
here = value of current cell
```

```
here(1,0) = value of next cell to right
here(0,1) = value of cell below current one
row(here) = row number of current cell

Note:
Although this is a useful function, it bypasses the minimal
recalc mechanism, so extensive use of it might make it
necessary to use Full Recalc.
```

1.82 Formula Function: HEX...

```
                HEX
Convert a number to a hexadecimal string.

Format:
HEX(x)

Input:
number: x

Output:
string: hexadecimal string equivalent of x

Example:
hex(1234) = "4d2"

See Also:

                HEXTO
```

1.83 Formula Function: HEXTO...

```
                HEXTO
Convert a hexadecimal string to a number.

Format:
HEXTO(str)

Input:
string:
  str (string with a hexadecimal number)

Output:
number:
  decimal value of hexadecimal number in str
0:
  if str does not look like a hexadecimal number

Example:
hexto("4d2") = 1234

See Also:
```

```
                HEX
```

1.84 Formula Function: HLOOK...

```
                HLOOK
Look up a value from a table with a row offset.

Format:
HLOOK(value to be looked up,range,row offset)

Input:
number:
  value to be looked up
range:
  range
number:
  row offset (limits: offset >= 0)

Output:
number:
  the table defined by 'range' is searched horizontally
  for the largest value that is not greater than the
  look-up value, then the row offset is used to look up
  another value vertically from the table, which is
  returned.

ERROR:
  if offset is negative or value not found in table

HLOOK() will also return a cell or range address if used as a
cell or range argument to another function.

See Also:

                VLOOK
```

1.85 Formula Function: HOUR...

```
                HOUR
Extract hour number from a serial time number.

Format:
HOUR(time)

Input:
time:
  a serial time number

Output:
number:
  the hour number (0 to 23) in the serial time number
```

```
Example:
hour(now) = 13 (if the time is 1 p.m.)

See Also:

                TIME
                ,
                MINUTE
                ,
                SECOND
                ,
                NOW
```

1.86 Formula Function: IF...

```
IF
Return one of two arguments based on value of an expression.

Format:
IF(expression,value if expression true,
  value if expression false)

Input:
expression:
  any valid formula expression
argument:
  number, string, cell or range address
argument:
  number, string, cell or range address

Output:
if expression is true (non-zero), 'value if expression true'
is returned, otherwise 'value if expression value' is returned

IF() will also return a cell or range address if used as a
cell or range argument to another function.

Examples:
=if(a1>12,100,500)

This will return 100 if a1 is bigger than 12, otherwise it will
return 500.

=if(a1<1000,a1,1000)

This will return the value of a1 if a1 is less than 1000,
otherwise it will return 1000.

=sum(if(a1>1,a1:a10,b1:b10))

This will return the sum of a1:a10 if a1 > 1, otherwise it
will return the sum of b1:b10
```

1.87 Formula Function: INDEX...

```
                INDEX
Get value of a cell indexed within a range.

Format:
INDEX(range,column offset,row offset)
INDEX(range,sheet offset,column offset,row offset)

Input:
range:
  range
number:
  sheet offset
number:
  column offset
number:
  row offset

Output:
value:
  value of a cell within the provided range indexed by
  the selected sheet, column and row offsets
ERROR:
  if any offset is outside provided range

Example:
index(a1:c10,0,0) = value of cell a1
index(a1:c10,1,0) = value of cell b1
index(a1:c10,0,1) = value of cell a2

See Also:

                CELL
```

1.88 Formula Function: INF...

```
                INF
Constant: Infinity

Format:
INF

Input:
None

Output:
number: Infinity

See Also:

                NINF
```

1.89 Formula Function: INT...

```
                INT
Get integer part of a number.

Format:
INT(x)

Input:
number: x

Output:
number: integer part of x

Example:
int(1.45) = 1
int(2.91) = 2

See Also:

                FRAC
```

1.90 Formula Function: IRR...

```
                IRR
Calculate Internal Rate of Return on a series of cash flows.

Format:
IRR(range of cash flows,guess)

Input:
range:
  cash flows
number:
  guess:  a number between 0 and 1 that represents a guess
  at the possible IRR.

Output:
number:
  the discount rate that would cause the net present
  value of a string of cash flows over a number of
  periods (defined by the number of cells in the range)
  to be equal to zero.

Example:
To calculate the Internal Rate of Return of a string of cash
flows in the range a1:c10 at a guess rate of 6%:

irr(a1:c10,6%)

Note:
IRR() is an iterative function.  It will attempt to find the
solution by successive approximation.  IRR() will make a
maximum of 100 attempts to arrive at a figure that is within
```

```
0.000001 accuracy.

See Also:

                NPV
                ,
                PV
                ,
                FV
```

1.91 Formula Function: ISERR...

```
                ISERR
Test expression for ERROR condition.

Format:
ISERR(expression)

Input:
expression: any valid formula expression

Output:
TRUE:
  if expression evaluated caused an ERROR
FALSE:
  if expression evaluated did not cause an ERROR

Example:
if(iserr(a1),0,a1)

This returns 0 if a1 is an ERROR, otherwise returns the value
of a1.

Note:
During evaluation of the expression in ISERR(), normal ERROR
checking and flagging is ignored.  This allows you to safely
trap ERROR conditions.

See Also:

                ERROR
                ,
                IZERR
                ,
                IF
```

1.92 Formula Function: ISINF...

```
                ISINF
Test expression for infinity.

Format:
```

```
ISINF(x)

Input:
number: x

Output:
TRUE:
  if x equals positive or negative infinity
FALSE:
  if x doesn't equal positive or negative infinity

Example:
if(isinf(a1),0,a1)

This returns 0 if a1 equal infinity, otherwise returns the
value of a1.

See Also:

                IF
```

1.93 Formula Function: ISNUM...

```
                ISNUM
Test for a number.

Format:
ISNUM(x)

Input:
number: x

Output:
TRUE:
  if x is a number or empty
FALSE:
  if x is a string

Example:
if(isnum(a1),a1,0)

This returns 0 if a1 is not a number, otherwise returns the
value of a1.

See Also:

                IF
                ,
                ISSTR
```

1.94 Formula Function: ISRANGE...

```
                ISRANGE
Test for a range.

Format:
ISRANGE(range)

Input:
range:  range

Output:
TRUE:
  if 'range' is a valid range
FALSE:
  if 'range' is no a valid range

Example:
IF(israngе(test),sum(test),0)

See Also:

                IF
```

1.95 Formula Function: ISSTR...

```
                ISSTR
Test for a string.

Format:
ISSTR(str)

Input:
string: str

Output:
TRUE:
  if str is a string
FALSE:
  if str is not a string

Example:
if(isstr(a1),a1,"default")

This returns "default" if a1 is not a string, otherwise returns
the value of a1.

See Also:

                IF
                ,
                ISNUM
```

1.96 Formula Function: IZERR...

```
                IZERR
Test expression for ERROR and Infinity conditions.

Format:
IZERR(expression)

Input:
expression:
  any valid formula expression

Output:
TRUE:
  if expression evaluated caused an ERROR, or was
  a negative or positive infinity

FALSE:
  if expression evaluated did not cause an ERROR and
  is not a negative or positive infinity

Example:
if(izerr(a1),0,a1)

This returns 0 if a1 is an ERROR or infinity, otherwise returns
the value of a1.

Note:
IZERR is provided for compatibility reasons with imported .WKS
sheets.  When importing .WKS sheets, ISERR() is automatically
converted to IZERR().

FinalCalc supports Infinity and will not cause an ERROR.
IZERR() is the only case when it is considered to be an ERROR.

During evaluation of the expression in IZERR(), normal ERROR
checking and flagging is ignored.  This allows you to safely
trap ERROR conditions.

See Also:

                ERROR
                ,
                ISERR
                ,
                IF
```

1.97 Formula Function: KEEP...

```
                KEEP
Remove all characters from a string except for specific
characters.

Format:
```

```
KEEP(str,template)

Input:
string: str
string: template

Output:
string:
  'str' with all characters that are not in 'template'
  removed.

Example:
keep("this is a test","tis") = "tisiststst"

See Also:

                REMOVE
```

1.98 Formula Function: LAST...

```
                LAST
Get last valid entry from multiple numeric or string arguments.

Format:
LAST(multiple range/cell/expression arguments)

Input:
One or more of:
 - numbers
 - cells
 - ranges

Output:
number or string:
  value of last valid entry from arguments

Example:
last(a1:b10) returns the value of the last cell in the range
a1:b10 that contains a valid value

See Also:

                FIRST
                ,
                PICK
                ,
                FLAST
                ,
                CLAST
```

1.99 Formula Function: LEFT...

```
                LEFT
Get part of the left side of a string.

Format:
LEFT(str,n)

Input:
string:
  str
number:
  n

Output:
string:
  the leftmost n characters from 'str'

Example:
left("this is a test",4) = "this"


LEFT
Value or address of cell to the left of the current cell.

Format:
LEFT

Input:
None

Output:
cell:
  address or value of cell left of current cell

Example:
=left+1    = cell to the left + 1
=col(left) = column number of cell to the left


See Also:

                RIGHT
                ,
                ABOVE
                ,
                BELOW
                ,
                MID
                ,
                LEN
```

1.100 Formula Function: LEN...

```
LEN
Get the length of a string.
```

```
Format:
LEN(str)

Input:
string:
  str

Output:
number:
  number of characters in 'str'.

Example:
len("this is a test") = 14
```

1.101 Formula Function: LIMIT...

```
LIMIT
Test a value against limits.

Format:
LIMIT(x,a,b,value if x<a,value if a<=x<=b,value if b<x)

Input:
number: x
number: a
number: b
number: value if x<a
number: value if a<=x<=b
number: value if b<x

Output:
number:
  returns one of three values depending on x, a, and b.
  a must be smaller than b, and they define the limits
  that value x is compared to.  If x is smaller than the
  limit, then value 1 (value if x<a) is returned.  If x
  is within a and b, then value 2 (value if a<=x<=b) is
  returned.   Else if x is larger than the limit, then
  value 3 (value if b<x) is returned.

ERROR:
  if a > b
```

1.102 Formula Function: LOG...

```
                LOG
Logarithm of a number to base 10.

Format:
LOG(x)
```

```
Input:
number:
  x (limit: x <= 0.0)

Output:
number:
  logarithm of x to base 10
ERRROR:
  if x exceeds limits

Example:
log(12) = 1.079181246

See Also:

                LOGN
```

1.103 Formula Function: LOGN...

```
                LOGN
Natural (base e) logarithm of a number.

Format:
LOGN(x)

Input:
number:
  x (limit: x <= 0.0)

Output:
number:
  natural logarithm of x (base e)
ERRROR:
  if x exceeds limits

Example:
logn(12) = 2.48490665

See Also:

                LOG
```

1.104 Formula Function: LOWER...

```
                LOWER
Convert a string to all lowercase characters.

Format:
LOWER(str)

Input:
string:
```

```
  str

Output:
string:
  lower-case version of str

Example:
=lower("TEST") = "test"

See Also:

                UPPER
                ,
                PROPER
```

1.105 Formula Function: MAX...

```
                MAX
Highest value of one or more numeric values.

Format:
MAX(multiple range/cell/expression arguments)

Input:
One or more of:
 - numbers
 - cells that contain numeric values
 - ranges (only cells with numeric contents in the range are used)

Output:
number:
  Highest numeric value of all arguments provided, or
  zero if no arguments were found.

Example:
max(a1:b10,12,100,a1/23))
max(100,a1,b10:c20,cos(12))

See Also:

                MIN
                ,
                CMAX
                ,
                FMAX
```

1.106 Formula Function: MID...

```
                MID
Get part of a string.

Format:
```

```
MID(str,start,length)

Input:
string:
  str
number:
  start
number:
  length

Output:
string:
  'length' (or less) characters from 'str' starting at
  position 'start'.

Example:
mid("this is a test",5,4) = "is a"

See Also:

                LEFT
                ,
                RIGHT
                ,
                LEN
```

1.107 Formula Function: MIN...

```
                MIN
Lowest value of one or more numeric values.

Format:
MIN(multiple range/cell/expression arguments)

Input:
One or more of:
 - numbers
 - cells that contain numeric values
 - ranges (only cells with numeric contents in the range are used)

Output:
number:
  Lowest numeric value of all arguments provided, or
  zero if no arguments were found.

Example:
min(a1:b10,12,100,a1/23))
min(100,a1,b10:c20,cos(12))

See Also:

                MAX
                ,
                CMIN
                ,
```

FMIN

1.108 Formula Function: MINUTE...

```
                MINUTE
Extract minute number from a serial time number.

Format:
MINUTE(time)

Input:
time:
  a serial time number

Output:
number:
  the minute number (0 to 59) in the serial time number

Example:
minute(now) = 12 (if the time is 10:12)

See Also:
```

TIME
,
HOUR
,
SECOND
,
NOW

1.109 Formula Function: MOD...

```
MOD
Get remainder of a division.

Format:
MOD(x,y)

Input:
number: x
number: y

Output:
number:
  value of x modulo y (same as x%y).  This is the
  remainder after dividing x over y.

Example:
mod(15,6) = 3 (remainder of 15/6)
```

1.110 Formula Function: MONTH...

```
                MONTH
Get month number from a serial date number.

Format:
MONTH(date)

Input:
date:
  serial date number

Output:
number:
  from 1 to 12, the month number from the provided serial
  date number

Example:
day(today) = 2  (if the current month was February)

See Also:
```

DATE
,
DAY
,
YEAR
,
TODAY
,
DATESTR
,
DATEVALUE

1.111 Formula Function: MONTHS...

```
MONTHS
Get number of months in a date.

Format:
MONTHS(date)

Input:
date:
  serial date number

Output:
number:
  the number of months in the date number provided.

Example:
MONTHS() is useful for comparisons with month counts:

months(date1)-months(date2)  returns the number of months
```

between date1 and date2.

1.112 Formula Function: NAMESTR...

NAMESTR
Convert cell or range definition to a string.

Format:
NAMESTR(cell/range)

Input:
cell or range definition

Output:
string:
  cell or range name string

Example:
namestr(a1)
  returns A1

namestr(a1:b10)
  returns A1:B10

namestr(cell(0,1))
  returns A1

namestr(a_a1)
  returns A_A1

See Also:

CELL
,
RANGE

1.113 Formula Function: NINF...

NINF
Constant: Negative infinity.

Format:
NINF

Input:
None

Output:
number: Negative Infinity

See Also:

INF

1.114 Formula Function: NOT...

```
NOT
Logical Not.

Format:
NOT(x)

Input:
number: x

Output:
FALSE:
  if x is TRUE (non-zero)
TRUE:
  if x is FALSE (zero)

Example:
if(not(a1),"non-zero","zero")
is the same as:
if(a1,"zero","non-zero")
```

1.115 Formula Function: NOW...

```
NOW
Returns the current date and time value.

Format:
NOW

Input:
None

Output:
the current date/time serial number.  The time/date serial
number is defined as the number of days since 1-Jan-1978.  The
integer value is the number of days, and the fraction is the
time.

Example:
if the date is 1-Feb-1994 and the time is 12:00 noon, NOW will
return 5875.5, where 5875 is the number of days since
1-Jan-1978 and .5 is half a day, thus 12:00 noon.

Note:
NOW is a live function, it is always evaluated at every
recalc.  If you only need the date number and not the time
number, use the TODAY function instead.

See Also:
```

TODAY
,
TIME
,
HOUR
,
MINUTE
,
SECOND
,
TIMESTR
,
TIMEVALUE

1.116 Formula Function: NPV...

NPV
Calculate the Net Present Value of a string of cash flows.

Format:
NPV(range of cash flows,discount rate)

Input:
range:
  cash flows
number:
  discount rate

Output:
number:
  the net present value of a string of cash flows over
  a number of periods (defined by the number of cells in
  the range) given the discount rate.

Example:
To calculate the net present value of a string of cash flows
in the range a1:c10 at a discount rate of 6%:

npv(a1:c10,6%)

See Also:

IRR
,
PV
,
FV

1.117 Formula Function: NUM...

```
                NUM
Get a number.

Format:
NUM(x)

Input:
number:
  x

Output:
number:
  if x a number, x is returned
0:
  if x is not a number

Example:
num(12) = 12
num("test") = 0
num(a1) = value of a1 if a1 is number, or 0 if a1 contains a
  string.

Note:
NUM is used where you want to make sure that a number is
provided to another function and not a string.

See Also:

                ISNUM
                ,
                STR
```

1.118 Formula Function: OCT...

```
                OCT
Convert a number to an octal string.

Format:
OCT(x)

Input:
number:
  x

Output:
string:
  octal string equivalent of x

Example:
oct(123) = "173"

See Also:

                OCTTO
```

1.119 Formula Function: OCTTO...

```
                OCTTO
Convert an octal string to a number.

Format:
OCTTO(str)

Input:
string:
  str

Output:
number:
  value of octal string in str

Example:
octto("173") = 123

See Also:

                OCT
```

1.120 Formula Function: PATTERN...

```
                PATTERN
Get the Graph Pattern number.

Format:
PATTERN(str)

Input:
string:
  str (string matching the name of a Graph Pattern)

Output:
number:
  number of pattern in graph patterns list, or 0 if it
  was not found.

Example:
pattern("checkered") = 7 (number of pattern 'checkered')

See Also:

                COLOR
                ,
                SYMBOL
```

1.121 Formula Function: PATH...

```
PATH
Get the full path name of a file, directory, or current project.

Format:
PATH()
PATH(directory)
PATH(filename)

Input:
string:
  directory (name of a valid directory)
string:
  filename (name of a valid file)

Output:
string:
  if a directory or file name was provided, returns the
  full path name to that directory or file, otherwise
  returns the full path name to the current project file.
  (If the current project does not have a file name yet,
  "Untitled" is returned)

Example:
path("C:list") = "Workench:c/list".
path() = "Work:FinalCalc/projects/demo.sheet"
```

1.122 Formula Function: PERM...

```
                PERM
Calculate the Permutation possibilities of a selection of
distinct items from a group.

Format:
PERM(n,r)

Input:
number: n
number: r

Output:
number:
  Permutation value of two numbers, n and r.  (as used in
  Probability and Statistics) which is usually defined as
  nPr = n!/(n-r)!

Example:
If you have a group of four people (A, B, C, D) and two jobs
to give out to only two of the four people, how many possible
permutations of Person/Job are possible?

perm(4,2) = 12

Specifically: AB, AC, AD, BA, BC, BD, CA, CB, CD, DA, DB, DC
(AB here means A gets Job 1 and B gets Job 2)
```

```
See Also:

                COMB
```

1.123 Formula Function: PI...

```
PI
Constant: PI value.

Format:
PI

Input:
None

Output:
number: PI = 3.141592654
```

1.124 Formula Function: PICK...

```
                PICK
Get a specific entry from one or more numeric or string
arguments.

Format:
PICK(offset,multiple cell/range/expression arguments)

Input:
number:
  offset, a positive or negative integer number.
  It is used to define which entry to select from
  the arguments provided.

  If offset is positive, it picks entries from the
  start of the arguments provided.  If it is negative,
  it picks entries from the end of the arguments
  provided.

     3  use the 3rd entry
     2  use the 2nd entry
     1  use the 1st entry
     0  use the last entry
    -1  use the entry before last
    -2  use the 2nd entry before last
    -3  use the 3rd entry before last

One or more of:
 - numbers
 - cells
 - ranges
```

```
Output:
number or string: value of first valid entry from arguments

Example:
  =pick(3,a1:a10)

  This will give you the value of the third cell in the range
  a1:a10, which is a3.

  =pick(-2,a1:a10)

  This will give you the value of the 2nd to last cell in the
  range a1:a10 which is a9.

See Also:

                LAST
                ,
                FIRST
                ,
                CPICK
                ,
                FPICK
```

1.125 Formula Function: PMT...

```
                PMT
Calculate loan repayment amount.

Format:
PMT(principal,interest,term of loan)

Input:
number: principal amount of loan
number: interest rate
number: term of loan

Output:
number:
  appropriate repayment amount for a loan given the
  principal, interest rate and term using compound interest

Example:
To calculate the yearly loan repayment amount for a loan of
$10,000, at a yearly interest rate of 12% for 15 years:

pmt(10000,12%,15) = $1468.24 per year

If the repayments were monthly, you should divided the yearly
interest rate by 12 and multiply the term by 12:

pmt(10000,12%/12,15*12) = $120.01 per month

Note the use of the % sign to indicate a percentage.  You can
use a normal number instead:  0.12 = 12%
```

```
See Also:

                PVA
                ,
                FVA
```

1.126 Formula Function: PROD...

```
PROD
Product of multiplying of one or more numeric values.

Format:
PROD(multiple range/cell/expression arguments)

Input:
One or more of:
 - numbers
 - cells that contain numeric values
 - ranges (only cells with numeric contents in the range are used)

Output:
number:
  Product of multiplying the numeric values of all
  arguments provided.

Example:
prod(a1:b10,12,100,a1/23))
prod(100,a1,b10:c20,cos(12))
```

1.127 Formula Function: PROPER...

```
                PROPER
Convert a string to proper case.

Format:
PROPER(str)

Input:
string:
  str

Output:
string:
  str with first character in every word in uppercase
  and the rest in lowercase

Example:
proper("this is a test.") = "This Is A Test"

See Also:
```

LOWER
,
UPPER

1.128 Formula Function: PV...

PV
Calculates the Present Value of an amount.

Format:
PV(future amount,interest rate,number of periods)

Input:
number: future amount
number: interest rate
number: periods

Output:
number:
  Present Value of a future amount with a specific
  number of periods and a specific interest rate using
  compound interest.

Example:
An amount of $10,000 in 15 years at an interest rate of 6% is
worth $4172.65 today. (with interest compounded yearly)

pv(10000,6%,15) = 4172.65

See Also:

FV
,
PVA
,
TERM
,
RATE

1.129 Formula Function: PVA...

PVA
Calculates the Present Value of a stream of fixed payments.

Format:
PVA(annuity amount,interest rate,number of periods)

Input:
number: annuity amount
number: interest rate
number: periods

```
Output:
number:
  Present Value of a stream of fixed payments over a
  number of periods presuming a specific interest rate
  using compound interest.

Example:
An annuity of $1,000 a month over 15 years at an interest rate
of 6% is worth  $118,503.51 today.

pva(1000,6%/12,15*12) = 118503.51

See Also:

                FVA
                ,
                PV
                ,
                TERMA
                ,
                RATEA
```

1.130 Formula Function: RADIANS...

```
                RADIANS
Convert a number from degrees to radians.

Format:
RADIANS(x)

Input:
number: x

Output:
number: value of x degrees in radians

Example:
degrees(114.592) = 12

See Also:

                DEGREES
```

1.131 Formula Function: RAND...

```
RAND
Get a random number.

Format:
RAND

Input:
```

```
None

Output:
number: a random number between 0.0 and 1.0

Example:
rand*15 = a random number between 0.0 and 15.0
```

1.132 Formula Function: RANGE...

```
                RANGE
Construct a range reference.

Format:
RANGE(col1,row1,col2,row2)
RANGE(sheet1,col1,row1,sheet2,col2,row)

Input:
sheet:
  [optional] sheet1
column:
  col1
row:
  row1
sheet:
  [optional] sheet2
column:
  col2
row:
  row2

Output:
range:
  a range that is constructed from the start and end
  sheets, columns and rows defined.

Example:
sum(range(0,1,3,10)) = sum(a1:d10)

Note:
RANGE() can only be used with a function expecting a range

See Also:

                RINDEX
                ,
                CELL
```

1.133 Formula Function: REGRES...

```
                REGRES
Perform Multivariate Regression Analysis.
```

```
Multivariate Regression Analysis analyzes a dependant variable
compared to a set of independant variables to find out the
relationship between them.

To use this REGRES() you need the following data:

1.  A set of observations from the dependant variable.  These
  should be set up as a single column of data in the sheet.

2.  One or more sets of observations from the independant
  variables.  These should be set up as one or more columns
  of data in the sheet.

See the Multivariate Regression Analysis section for more
details.

Format:
REGRES(indepenant variables range,dependant variable range,
  item,offset)

Input:
range:
  independant variables range
range:
  dependant variable range

string:
  'item' controls what analysis is performed on the data
  ranges.  This must be one of the following:

    R
      Get Multiple R
    R2
      Get Multiple R$^2$
    FV
      Get F Value
    FS
      Get F Significance
    MEAN
      Get Y Mean
    ERROR_I
      Get Standard Error of Intercept
    TEST_I
      Get Test for Hypothesis: I = 0
    SIG_I
      Get Signifance of TEST_I
    REG_SUM_SQUARES
      Get Sum of Squares of Regression
    RES_SUM_SQUARES
      Get Sum of Squares of Residual
    TOT_SUM_SQUARES
      Get Total Sum of Squares
    REG_FREEDOM
      Get Degrees of Freedom of Regression
    RES_FREEDOM
      Get Degrees of Freedom of Residual
```

```
    TOT_FREEDOM
      Get Total Degrees of Freedom
    REG_MEAN
      Get Mean Square of Regression
    RES_MEAN
      Get Mean Square of Residual
    RES_ERROR
      Get Standard Error of Estimate
    EST_I
      Get Estimated Intercept

    The following items require 'offset' to be defined:

    EST_Y
      Get Estimated Yn
    EST_X
      Get Estimated Xn value
    ERROR_X
      Get Standard Error for Xn
    TEST_X
      Get Test for Hypothesis: Xn = 0
    SIG_X
      Get Significance of TEST_X for Xn

number:
  [optional] offset
  Define the entry from a list to select.  This only
  applies to some items (as described above.)

  If 'offset' is outside the available data, zero is returned.

  If offset is not defined for an item that requires it, offset defaults to 0.

Output:
number:
  required item, according to 'item' and 'offset' above.
ERROR:
  in the following cases:
  - The Dependant Variable Range contains one data column
  - The number of rows in Dependant and Independant Variables
    don't match."
  - Insufficient number of data points
  - Unknown item name."

Example:
regres(a1:c10,d1:d10,"r2")    Get Multiple R$^2$
regres(a1:c10,d1:d10,"est_x",1) Get Estimated X1

See Also:

                FSIGN
                ,
                TSIGN
```

1.134 Formula Function: RATE...

```
                RATE
Calculate the interest rate of an investment.

Format:
RATE(future value,present value,periods)

Input:
number: future value
number: present value
number: periods

Output:
number:
  the interest rate of an investment, given its present
  and future values and the number of periods using
  compound interest

Example:
An amount of $10,000 today, would need an interest rate of
7.60% to be worth $30,000 in 15 years.  (with interest
compounded yearly)

rate(30000,10000,15) = 7.60%

If interest is compounded monthly, then divide the interest
rate by 12, and multiply the periods by 12.

See Also:

                TERM
                ,
                FV
                ,
                PV
```

1.135 Formula Function: RATEA...

```
                RATEA
Calculate the interest rate on an annuity, given its future
value, present value and number of periods.

Format:
RATEA(annuity,future value,periods)

Input:
number: annuity
number: future value
number: periods

Output:
number:
  the interest rate of a stream of fixed payments, given
```

```
  the future value of the annuity and future values and
  the number of periods using compound interest

See Also:

                TERMA
                ,
                FVA
                ,
                PVA
```

1.136 Formula Function: REMOVE...

```
                REMOVE
Remove specific characters from a string.

Format:
REMOVE(str,template)

Input:
string: str
string: template

Output:
string:
  'str' with all characters that are in 'template' removed.

Example:
remove("abcde 12345","cd34") = "abe 125"

See Also:

                KEEP
```

1.137 Formula Function: REPEAT...

```
REPEAT
Repeat a string multiple times.

Format:
REPEAT(str,n)

Input:
string:
  str
number:
  n

Output:
string:
  'str' repeated n time, to a maximum of 2047 characters.
```

```
Example:
repeat("test ",3) = "test test test "
```

1.138 Formula Function: REPLACE...

```
REPLACE
Replace a sub-string with another.

Format:
REPLACE(str1,x,y,str2)

Input:
string:
  str1
number:
  x
number:
  y
string:
  str2

Output:
string:
  y characters from 'str1' are deleted starting at
  position x and 'str2' is inserted in their place

Example:
replace("nice test",5,4,"TEST") = "NICE TEST"
```

1.139 Formula Function: REVERSE...

```
REVERSE
Reverse a string.

Format:
REVERSE(str)

Input:
string: str

Output:
string: reversed string

Example:
=lower("a test") = "tset a"
```

1.140 Formula Function: RIGHT...

```
                RIGHT
Get part of the right side of a string.

Format:
RIGHT(str,n)

Input:
string:
  str
number:
  n

Output:
string:
  the rightmost n characters from 'str'

Example:
right("this is a test",6) = "a test"


RIGHT
Value or address of cell to the right of the current cell.

Format:
RIGHT

Input:
None

Output:
cell:
  address or value of cell right of current cell

Example:
=right+1    = cell to the right + 1
=col(right) = column number of cell to the right


See Also:

                LEFT
                ,
                ABOVE
                ,
                BELOW
                ,
                MID
                ,
                LEN
```

1.141 Formula Function: RINDEX...

```
                RINDEX
Define a sub-range from a larger range.
```

```
Format:
RINDEX(base range,start column offset,start row offset,
  end column offset,end row offset)

RINDEX(base range,start sheet offset,start column offset,
  start row offset,end sheet offset,end column offset,
  end row offset)

Input:
range:  base range
number: [optional] start sheet offset
number: start column offset
number: start row offset
number: [optional] end sheet offset
number: end column offset
number: end row offset

Output:
range:
  a range built from offsets within the base range
ERROR:
  if offsets are outside the base range

Example:
rindex(C10:F100,0,0,0,0) = C10:C10
rindex(C10:F100,0,0,1,0) = C10:D10
rindex(C10:F100,1,0,3,0) = D10:F10
rindex(C10:F100,0,1,0,5) = C11:C15

Note:
There are major benefits of using RINDEX() over RANGE():

  1.  Safety:
    RINDEX() performs bounds checking for you and alerts you
    with ERROR if you have built a range that is outside the
    base range.  RANGE() has no safety net.

  2.  Speed:
    Formulas with RANGE() are live, i.e. calculated every time
    even if they haven't changed.  Formulas with RINDEX() are
    only calculated when a cell in the base range changes.
    This can make a big difference in speed in large sheets.

See Also:

                INDEX
                ,
                RANGE
```

1.142 Formula Function: ROMAN...

```
ROMAN
Convert a number to a Roman Numerals string.
```

```
Format:
ROMAN(x)

Input:
number:
  x  (from 1 to 100,000)

Output:
string:
  Roman Numerals string equivalent of x.

Example:
roman(1993) = "MCMXCIII"
```

1.143 Formula Function: ROUND...

```
ROUND
Round a number.

Format:
ROUND(x,y)

Input:
number: x
number: y

Output:
number: x rounded down to y digits

Example:
round(1234.56789,4)     1234.5679
round(1234.56789,3)     1234.5680
round(1234.56789,2)     1234.5700
round(1234.56789,1)     1234.6000
round(1234.56789,0)     1235.0000
round(1234.56789,-1)    1230.0000
round(1234.56789,-2)    1200.0000
round(1234.56789,-3)    1000.0000
```

1.144 Formula Function: ROW...

```
                ROW
Get row number of a cell or range.

Format:
ROW()
ROW(cell)
ROW(range)
ROW(range,cell)

Input:
cell:
```

```
  [optional] a cell name
range:
  [optional] a range name

Output:
Depends on arguments provided:

ROW()
returns the row number of the current cell.

ROW(cell)
returns the row number of the provided cell.

ROW(range)
returns the row number of the top left cell in the provided
range.

ROW(range,cell)
returns the row offset of 'cell' in 'range', or ERROR if
'cell' is outside 'range'.

Example:
row() = row number of current cell
row(c1) = 1
row(c2:d3) = 2
row(c1:e10,d3) = 2 (offset of row 3 to row 1)

See Also:

                ROWS
                ,
                COL
```

1.145 Formula Function: ROWS...

```
                ROWS
Get number of rows in a range.

Format:
ROWS(range)

Input:
range:
  range

Output:
number:
  number of rows in 'range'

Example:
rows(a1:a10) = 10

See Also:

                COLS
```

,
SHEETS

1.146 Formula Function: SECOND...

```
                SECOND
Extract second number from a serial time number.

Format:
SECOND(time)

Input:
time:
  a serial time number

Output:
number:
  the second number (0 to 59) in the serial time number

Example:
second(now) = 23 (if the time is 10:12:23)

See Also:
```

TIME
,
HOUR
,
MINUTE
,
NOW

1.147 Formula Function: SETCOLOR...

```
                SETCOLOR
Set a cell or range to a color number.

Format:
SETCOLOR(cell/range,color)

Input:
cell/range:
  cell or range specification

number:
  color number:
     0  default
     1  white
     2  red
     3  green
     4  blue
     5  orange
```

```
     6  pink
     7  cyan
     8  yellow

Output:
None

Example:
This function does NOT return any result, only zero.  So it can
be safely added to the end of any formula:

=A1+SETCOLOR(HERE,IF(A1<0,2,1))
This sets the current cell to the value of A1, and its color to
white.  If the number is less than 0, it sets the color to red.

Note:
This function may actually allocate new cells!  Use it with
caution.

See Also:

                GETCOLOR
                ,
                SETSTYLE
```

1.148 Formula Function: SETSTYLE...

```
                SETSTYLE
Set a cell or range to a specific style.

Format:
SETSTYLE(cell/range,color)

Input:
cell/range:
  cell or range specification

number:
  style number:
     0  normal
     1  bold
     2  italic
     4  undeline
     8  reverse
    16  shaded

    Combined styles are added together:
    bold+italic  = 1+2  = 3

Output:
None

Example:
This function does NOT return any result, only zero.  So it can
be safely added to the end of any formula:
```

```
=A1+SETSTYLE(HERE,IF(A1<0,9,1))
This sets the current cell to the value of A1, and its style to
bold.  If the number is less than 0, it sets the style to
bold-reverse.

Note:
This function may actually allocate new cells!  Use it with
caution.

See Also:

                GETSTYLE
                ,
                SETCOLOR
```

1.149 Formula Function: SHEET...

```
                SHEET
Get sheet number of a cell or range.

Format:
SHEET()
SHEET(cell)
SHEET(range)
SHEET(range,cell)

Input:
cell:
  [optional] a cell name
range:
  [optional] a range name

Output:
Depends on arguments provided:

SHEET()
returns the sheet number of the current cell.

SHEET(cell)
returns the sheet number of the provided cell.

SHEET(range)
returns the sheet number of the top left cell in the provided
range.

SHEET(range,cell)
returns the sheet offset of 'cell' in 'range', or ERROR if
'cell' is outside 'range'.

Example:
sheet() = sheet number of current cell
sheet(c_c1) = 3 (sheet C)
sheet(c_c1:e_d1) = 3 (sheet C)
sheet(a_c1:z_e10,b_d1) = 1 (offset of sheet B to sheet A)
```

```
See Also:

                SHEETS
                ,
                COL
                ,
                ROW
```

1.150 Formula Function: SHEETS...

```
                SHEETS
Get the number of sheets in a range.

Format:
SHEETS(range)

Input:
range:
  a range name

Output:
number:
  number of sheets in range

Example:
cols(a_a1:c_z10) = 3

See Also:

                ROWS
                ,
                COLS
```

1.151 Formula Function: SIGN...

```
SIGN
Test number for positive, negative or zero.

Format:
SIGN(x)

Input:
number: x

Output:
1:  if x > 0
0:  if x = 0
-1: if x < 0

Example:
if(sign(a1)=-1,"negative","positive")
```

1.152 Formula Function: SIN...

```
SIN
Sine of a number.

Format:
SIN(x)

Input:
number:
  x (limit: x < 6.7465e+9)

Output:
number:
  sine of x
ERROR:
  if x exceeds limits

Example:
sin(12) = -0.536572918
```

1.153 Formula Function: SINH...

```
SINH
Hyperbolic sine of a number.

Format:
SINH(x)

Input:
number:
  x (limit: x <= 709.778)

Output:
number:
  hyperbolic sine of x
ERROR:
  if x exceeds limits

Example:
sinh(12) = 81377.395
```

1.154 Formula Function: SLN...

```
                SLN
Calculate the yearly straight-line depreciation amount of a
fixed asset.

Format:
SLN(cost,salvage value,life)

Input:
```

```
number: cost
number: salvage value
number: life (in years)

Output:
number:
  the yearly straight-line depreciation amount of a fixed
  asset taking into consideration its salvage value at
  the end of its life.  This is defined as:

    SLN = (cost - salvage value)/life

Example:
If an asset is worth $50,000, and is expected to last 7 years
and to be worth $10,000 at the end of its lifetime, the yearly
depreciation amount should be $5,714.28.

sln(50000,10000,7) = 5714.28

See Also:

                SYD
```

1.155 Formula Function: SQRT...

```
SQRT
Square root of a number.

Format:
SQRT(x)

Input:
number:
  x (limits: x >= 0.0)

Output:
number:
  square root of x

Example:
sqrt(4) = 2
```

1.156 Formula Function: STD...

```
                STD
Standard deviation of one or more numeric values.

Format:
STD(multiple range/cell/expression arguments)

Input:
One or more of:
```

```
 - numbers
 - cells that contain numeric values
 - ranges (only cells with numeric contents in the range are used)

Output:
number:
  Standard deviation of all arguments provided.

Example:
std(a1:b10,12,100,a1/23))
std(100,a1,b10:c20,cos(12))

See Also:

                COUNT
                ,
                AVG
                ,
                SUM
                ,
                STDS
```

1.157 Formula Function: STDS...

```
                STDS
Standard deviation of a sample population of one or more
numeric values.

Format:
STDS(multiple range/cell/expression arguments)

Input:
One or more of:
 - numbers
 - cells that contain numeric values
 - ranges (only cells with numeric contents in the range are used)

Output:
number:
  Standard deviation of a sample population of all
  arguments provided.

Example:
stds(a1:b10,12,100,a1/23))
stds(100,a1,b10:c20,cos(12))

See Also:

                COUNT
                ,
                AVG
                ,
                SUM
                ,
```

```
                STD
```

1.158 Formula Function: STR...

```
                STR
Get a string.

Format:
STR(str)

Input:
string: str

Output:
string:
  if str is a string, str is returned, otherwise a null
  string ("") is returned.

Example:
len(str(a1))  This makes sure that LEN() always gets a string
argument.  Without STR(), if a1 was a number, LEN() would cause
an ERROR.

Note:
STR is used where you want to make sure that a string is
provided to another function and not a number.

See Also:

                NUM
```

1.159 Formula Function: STRING...

```
                STRING
Convert a number to a string with fixed number of decimals.

Format:
STRING(x,n)

Input:
number: x    number to convert
number: n    number of decimals places

Output:
string: a string with the value of x

Example:
string(12.345,2) = "12.34"
string(12.345,5) = "12.34500"

See Also:
```

DEC
,
VALUE

1.160 Formula Function: SUM...

SUM
Sum of one or more numeric values.

Format:
SUM(multiple range/cell/expression arguments)

Input:
One or more of:
 - numbers
 - cells that contain numeric values
 - ranges (only cells with numeric contents in the range are used)

Output:
number:
  Sum of all arguments provided.

Example:
sum(a1:b10,12,100,a1/23))
sum(100,a1,b10:c20,cos(12))

See Also:

COUNT
,
AVG
,
CSUM
,
FSUM

1.161 Formula Function: SYD...

SYD
Calculate the yearly depreciation amount of a fixed asset for a
number of periods using the sum-of-the-years'-digits method.

Format:
SYD(cost,salvage value,life,period)

Input:
number: cost
number: salvage value
number: life (in years)
number: period (in years)

Output:

```
number:
  the depreciation amount for the selected period of a
  fixed asset for a number of periods, taking into
  consideration its salvage value at the end of its life,
  using the sum-of-the-years'-digits method.  This method
  has an accelerated deprecation in the early years of an
  asset's life.

See Also:

                SLN
```

1.162 Formula Function: SYMBOL...

```
                SYMBOL
Get the Graph Symbol number.

Format:
SYMBOL(str)

Input:
string:
  str (string matching the name of a Graph Symbol)

Output:
number:
  number of symbol in graph symbols list, or 0 if it
  was not found.

Example:
pattern("square") = 4 (number of pattern 'square')

See Also:

                COLOR
                ,
                PATTERN
```

1.163 Formula Function: TAN...

```
                TAN
Tangent of a number.

Format:
TAN(x)

Input:
number: x

Output:
number: tangent of x
```

```
Example:
tan(12) = -0.636

See Also:

                SIN
                ,
                COS
                ,
                TANH
                ,
                ATAN
                ,
                ATAN2
```

1.164 Formula Function: TANH...

```
                TANH
Hyperbolic tangent of a number.

Format:
TANH(x)

Input:
number: x

Output:
number: hyperbolic tangent of x

Example:
tanh(1) = 0.762

See Also:

                SIN
                ,
                COS
                ,
                TAN
                ,
                ATAN
                ,
                ATAN2
```

1.165 Formula Function: TERM...

```
                TERM
Calculate the number of periods required to reach a certain
future value.

Format:
TERM(interest rate,future value,present value)
```

```
Input:
number: rate of interest
number: future value
number: present value

Output:
number:
  number of periods required to reach the specified future
  value, given the interest rate and the present value.

Example:
An amount of $10,000 today, would need 14.275 periods at an
interest rate of 8% to be worth $30,000 in 15 years.  (with
interest compounded yearly)

term(8%,30000,10000) = 14.275

If interest is compounded monthly, then divide the interest
rate by 12, and multiply the periods by 12.

See Also:

               FV
               ,
               PV
               ,
               RATE
```

1.166 Formula Function: TERMA...

```
               TERMA
Calculate the number of periods required to reach a certain
future value for a stream of fixed payments.

Format:
TERMA(annuity,interest rate,future value)

Input:
number: annuity
number: interest rate
number: future value

Output:
number:
  number of periods required to reach the specified
  future value for a stream of fixed payments at a
  specific rate of interest

Example:
A stream of payments of $1000 each would need 28.55 periods
to be worth $100,000 presuming an interest rate of 8%.
(with interest compounded yearly)

terma(1000,8%,100000) = 28.55
```

```
If interest is compounded monthly, then divide the interest
rate by 12, and multiply the periods by 12.

See Also:

                FVA
                ,
                PVA
                ,
                RATEA
```

1.167 Formula Function: THERE...

```
                THERE
Cell address of cursor.

Format:
THERE

Input:
None

Output:
cell:
  address of cell where the cursor is currently pointing

Example:
col(there) = column where the cursor currently is

See Also:

                HERE
```

1.168 Formula Function: TIME...

```
                TIME
Calculate a serial time number from time components.

Format:
TIME(hour,minute,second)

Input:
number: hour (0 to 23)
number: minute (0 to 59)
number: second (0 to 59)

Output:
time:
  serial time number calculated from hour, minute and
  second arguments
```

```
Example:
time(12,0,0) = 0.5

See Also:

                TIMESTR
                ,
                TIMEVALUE
                ,
                HOUR
                ,
                MINUTE
                ,
                SECOND
                ,
                NOW
```

1.169 Formula Function: TIMESTR...

```
                TIMESTR
Convert a time number to a string using a custom format.

Format:
TIMESTR(str,time)

Input:
string:
  str
time:
  time

Output:
string:
  convert time to a string using the string format
  provided in 'str'

Example:
timestr("h:mm aa",NOW) = "12:10 am"

See Also:

                TIME
                ,
                TIMEVALUE
                ,
                HOUR
                ,
                MINUTE
                ,
                SECOND
                ,
                NOW
```

1.170 Formula Function: TIMEVALUE...

```
                TIMEVALUE
Convert a string from text to a time number.

Format:
TIMEVALUE(str)

Input:
string: str

Output:
time:
  the time number in 'str'.  This must contain a string
  that looks like a time number, e.g. "12:00" or "12:10:00".

ERROR:  if str doesn't look like a time number

Example:
timevalue("12:00") = 0.5

See Also:

                TIME
                ,
                TIMESTR
                ,
                HOUR
                ,
                MINUTE
                ,
                SECOND
                ,
                NOW
```

1.171 Formula Function: TODAY...

```
                TODAY
Current system date.

Format:
TODAY

Input:
None

Output:
date:
  serial date number for the current date.  The date
  serial number is defined as the number of days since
  1-Jan-1978.

Example:
if the date is 1-Feb-1994, TODAY will return 5875, where 5875
```

```
is the number of days since 1-Jan-1978.

Note:
TODAY is not a live function, it is only evaluated:
  a.  if the cell has changed.
  b.  if the current system date has changed since the last
    recalc.
  c.  when the project is first loaded.

See Also:

               NOW
               ,
               DATE
               ,
               YEAR
               ,
               MONTH
               ,
               DAY
```

1.172 Formula Function: TRANSLATE...

```
TRANSLATE
Convert characters in a string.

Format:
TRANSLATE(str,template1,template2)

Input:
string: str
string: template1
string: template2

Output:
string:
  'str' is copied as is except for any character that is
  found in 'template1' is replaced with the corresponding
  character from 'template2'.

Example:
translate("this is a test","th","TH") = "THis is a TesT"
translate("this is a test","th","12") = "12is is a 1es1"
```

1.173 Formula Function: TRIM...

```
TRIM
Clean multiple spaces from a string.

Format:
TRIM(str)
```

```
Input:
string: str

Output:
string:
  contents of str except that multiple spaces are converted
  into a single space.  Trailing spaces are also removed.

Example:
trim("this    is   a    test   ") = "this is a test"
```

1.174 Formula Function: TRUE...

```
                TRUE
Constant: value of TRUE (1).

Format:
TRUE

Input:
None

Output:
TRUE (1)

See Also:

                FALSE
```

1.175 Formula Function: TSIGN...

```
                TSIGN
Get Significance of T Value given the degrees of freedom.
(used for Multivariate Regression Analysis)

Format:
TSIGN(f value,degrees of freedom)

Input:
number: t value
number: degrees of freedom

Output:
number:
  Significance of T value given the provided degrees of
  freedom.

See Also:

                REGRES
                ,
                FSIGN
```

1.176 Formula Function: UPPER...

```
                UPPER
Convert a string to uppercase characters.

Format:
UPPER(str)

Input:
string: str

Output:
string: converts str to all uppercase characters

Example:
upper("test") = "TEST"

See Also:

                LOWER
                ,
                PROPER
```

1.177 Formula Function: VALUE...

```
                VALUE
Convert a string to a number.

Format:
VALUE(expression)

Input:
string:
  expression, any valid formula or math expression

Output:
result:
  number or string depending on result of evaluated
  expression
ERROR:
  if expression was not valid

Example:
evaluate("sin(12)") = sine of 12

Note:
You can use cell and range names in 'expression' but it is not
recommended since VALUE() is not a live function.  If you need
to use cell or range names, use the EVALUATE() function
instead.

See Also:

                EVALUATE
```

,
STRING
,
DEC

1.178 Formula Function: VAR...

VAR
Variance of one or more numeric values.

Format:
VAR(multiple range/cell/expression arguments)

Input:
One or more of:
 - numbers
 - cells that contain numeric values
 - ranges (only cells with numeric contents in the range are used)

Output:
number:
  Variance of all arguments provided

Example:
var(a1:b10,12,100,a1/23))
var(100,a1,b10:c20,cos(12))

See Also:

VARS
,
SUM
,
COUNT
,
AVG

1.179 Formula Function: VARS...

VARS
Variance of a sample population of one or more numeric values.

Format:
VARS(multiple range/cell/expression arguments)

Input:
One or more of:
 - numbers
 - cells that contain numeric values
 - ranges (only cells with numeric contents in the range are used)

Output:

```
number:
  Variance of a sample population of all arguments provided

Example:
vars(a1:b10,12,100,a1/23))
vars(100,a1,b10:c20,cos(12))

See Also:

                VAR
                ,
                SUM
                ,
                COUNT
                ,
                AVG
```

1.180 Formula Function: VLOOK...

```
                VLOOK
Look up a value from a table with a column offset.

Format:
VLOOK(value to be looked up,range,column offset)

Input:
number:
  value to be looked up
range:
  range
number:
  column offset (limits:  offset >= 0)

Output:
number:
  the table defined by 'range' is searched vertically
  for the largest value that is not greater than the
  look-up value, then the column offset is used to look
  up another value horizontally from the table, which is
  returned.

ERROR:
  if offset is negative or value not found in table

VLOOK() will also return a cell or range address if used as a
cell or range argument to another function.

See Also:

                HLOOK
```

1.181 Formula Function: WINDCHILL...

```
WINDCHILL
Calculate the wind chill temperature.

Format:
WINDCHILL(temperature,wind speed)

Input:
number:
  temperature (in degrees Celcius)
number:
  wind speed (in kilometers per hour)

Output:
number:
  wind chill factor (in degress Celcius) for the
  temperature and wind speed specified

Example:
If the current temperature is 11 degress Celcius and the wind
speed is 20 kilometers per hour, the wind chill temperature is
4.83 degress Celcius

windchill(11,20) = 4.83
```

1.182 Formula Function: WORD...

```
                WORD
Get a specific word from a string.

Format:
WORD(str,n)

Input:
string:
  str
number:
  n

Output:
string:
  the n'th word from 'str'.  If n is more than the number
  of words in 'str', a null string ("") is returned.

Example:
word("this is a test",2) = "is"

Note:
A word is defined as a group of alphanumeric characters
separated from another 'word' by one or more non-alphanumeric
characters.

See Also:

                WORDS
```

1.183 Formula Function: WORDS...

```
                WORDS
Get the number of words in a string.

Format:
WORDS(str)

Input:
string:
  str

Output:
number:
  the number of words in string 'str'

Example:
words("this is a test") = 4

Note:
A word is defined as a group of alphanumeric characters
separated from another 'word' by one or more non-alphanumeric
characters.

See Also:

                WORD
```

1.184 Formula Function: YEAR...

```
                YEAR
Get year number from a serial date number.

Format:
YEAR(date)

Input:
date:
  serial date number

Output:
number:
  the year number from the provided serial date number

Example:
day(today) = 1994  (if the current year was 1994)

See Also:

                DATE
                ,
                DAY
                ,
                MONTH
```

```
,
TODAY
,
DATESTR
,
DATEVALUE
```