Spreadsheet Manual

Ensign 10

Table of Contents

Ensign Spreadsheet

Spreadsheet

Click the Main | Spreadsheet ribbon button to display a spreadsheet. The spreadsheet can be used for showing text, variable values, and results from spreadsheet formula.

Features

- Simple formula editing interface
- Auto recalculation
- Single cell recalculation, full recalculation
- Extensive range of mathematical functions
- Save with formulas
- Single cell references in formulas
- Cell range formulas
- Formula precision for grid on cell basis
- Display formulas or formula results
- Date / time functions
- Intelligent formula aware copy and paste
- Extended with Ensign custom functions
- Cell name mode is the A1-style
- Intelligent and customizable hints while editing formulas
- HTML tags for cell formatting and visual effects

Overview

This is an overview of using cell references and built-in functions. Formulas can contain cell references, constants, single parameter functions, multiple parameters functions, cell name references and cell range functions.

Cell References

Cell references use the A1 format which consists of 2 parts: the column identifier and the row identifier. The column identifier is a character, starting from A for the first column, B for the second column, etc. The row identifier starts at 1 for the first editable row.

Example: The cell in column 1, row 1 is A1. The cell in column 10, row 25 is J25

Cell formulas are relative by default. That means that when cell formulas are involved in copy & paste operations or row/column insert and delete, the spreadsheet will automatically adapt the formulas to address the proper relative cells. Absolute cell addresses will not be modified during clipboard copy & paste operations or during row/column insert and delete. Prefix the

cell address row or column part with '$' to indicate an absolute cell address.

Example:

A$1 : A is a relative column address, 1 is an absolute row address
$B$2 : B is an absolute column address, 2 is an absolute row address

Cell Ranges

Cell ranges are identified by top left cell and bottom right cell split by ':' As such, the first 15 cells in column 1, can be specified as A1:A15.

Example:

A1:B3 : specifies the range of cells from cell 1,1 to cell 2,3
$A$1:$B$3 : specifies an absolute cell range from cell 1,1 to cell 2,3

Cell Editing

Numbers and text can be typed directly into a cell. Text may include HTML tags, such as text to Bold the text.

	A	B	C	D
1	Normal	**Bold**	Size 12	Red
2	*Italic*	Underline	***All 3***	Green

The A1 cell entry is just the text Normal.

The B1 cell entry is Bold.

The C1 cell entry is Size 12

The D1 cell entry is Red or Red

The A2 cell entry is Italic

The B2 cell entry is <u>Underline</u>

The C2 cell entry is <u>All 3</u>

The D2 cell entry is <body bgcolor=”clBlack”>Green</body>

Functions and formula are entered by beginning the cell entry with the '=' character.

Examples: =Sum(A1:A10) =IF(V(100),”Above”,”Below”)

Function names are not case sensitive. Sum, sum, and SUM are all equivalent. The =IF example shows the Variable function being used as the 1st parameter of the IF function.

Formula Formatting

Use the Concatenate function to wrap a function with HTML formatting tags.

This example shows in Red the value of variable 100.

```
=concatenate("<font color=&quot;clRed&quot;>",V(100),"</font>")
```

The next example calculates a percent change for a symbol and appends the % character.

```
=concatenate(100*Net("ES #F","ES")/Yesterday("ES #F","ES"),"%")
```

This complex example uses the IF function to display a quote value in Blue or in Red.

```
=if(V(1),concatenate("<font color=&quot;clBlue&quot;>",last("NQ #F","ES"),"</font>"),
        concatenate("<font color=&quot;clRed&quot;>",last("NQ #F","ES"),"</font>"))
```

Show Formula 

After entering a function, press the Enter key. The cell will display the function result. Click in a cell to select it. Click a 2nd time to redisplay the cell's function or formula, or click the toolbar button to Show Formula. Click the button a 2nd time to Hide Formula.

	A	B
1		2
2		5.6
3		8
4	Sum=	=Sum(B1:B3)

	A	B
1		2
2		5.6
3		8
4	Sum=	15.6000

Select a cell and press Backspace to erase its contents.

Variables

Ensign has an array of global variables with indexes from 0 to 199 that can be used to exchange values between the spreadsheet and other studies. The spreadsheet can read values using the =V(index) or the =Get(index) functions. Both functions are identical.

The spreadsheet can write values to the array using the =Set(index,value) function.

Assume that cell B2 has the value 5.6. This value can be written to V(100) using =Set(100,B2) as a function placed in any unused cell. Now V(100) = 5.6.

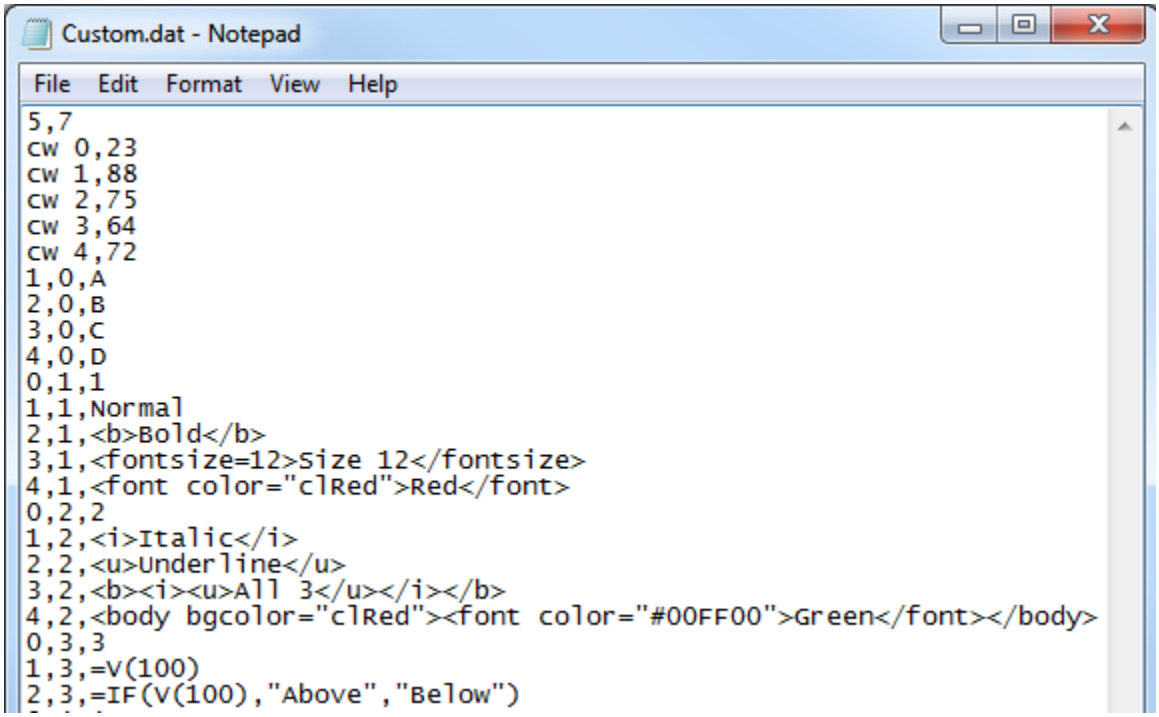
The DYO study and the ESPL programming language have statements that can read and write spreadsheet cells.

Manual Calculate 

Click the Calculate button to recalculate the spreadsheet. See the Calculate Mode setting on the Properties form for enabling automatic calculation.

File Content

The spreadsheet cell definitions are saved with the file name as entered in the list box on the top of the form, and have '.dat' as the extension. Spreadsheet files are saved in the C:\Ensign Software\Spread folder. Example file content:

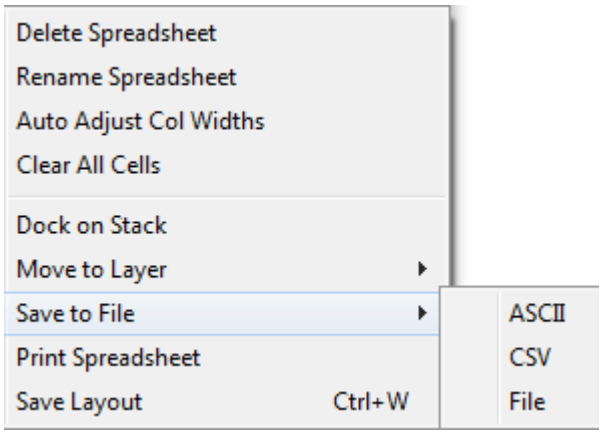


Save to File

The display values of the spreadsheet can be exported to 3 file types. Right mouse click on the spreadsheet to show the pop-up menu. Select the Save to File menu.

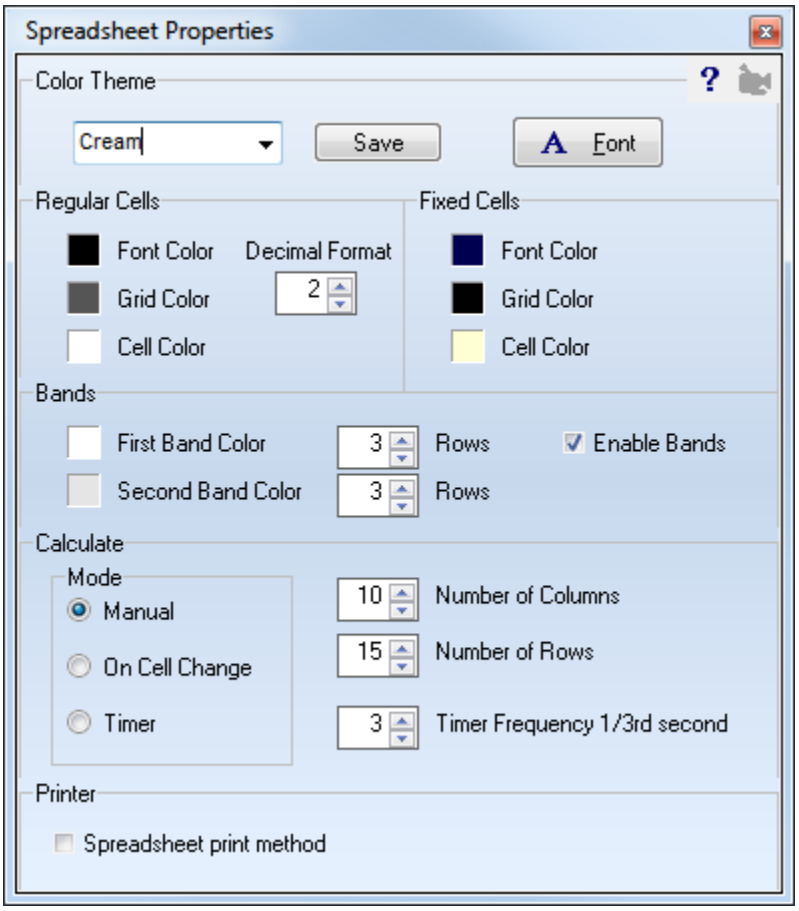
The Save to File menus will write files in the C:\Ensign10\Spread folder, using the name of the spreadsheet with different file extensions.

- ASCII – extenstion is .txt
- CSV – extension is .csv
- File – suffix appended is -file.txt



Properties

The colors and controls for the spreadsheet can be customized on the properties form.



Theme - Several color themes have been provided. Select a theme from the Color Theme list box. A theme can be adapted to personal preference, and then saved by clicking the Save button. Entering a new name and clicking the Save button will create a new theme file.

Colors - Click on a square color box to select a color.

Decimal Format – This property sets the number of decimals to show for formatting floating point numbers.

Calculate Mode – This frame controls when the spreadsheet calculates. Use the Manual selection when the spreadsheet is quite static, such as being used to create a report, or during the design of the sheet's logic. Click the Calculate button on the spreadsheet toolbar to manually recalculate the spreadsheet.

The On Cell Change will recalculate the sheet when a cell entry changes.

Use the Timer selection to have the sheet automatically recalculate. This would be the setting to use to update the sheet with quote prices, study values, and formula that are calculating values to be plotted.

Timer Frequency – This setting is used with the Timer calculate mode. Enter 3 to have the spreadsheet recalculate once per second. Enter 15 to have the spreadsheet recalculate once every 5 seconds.

Number of Columns, Number of Rows – Change these properties to adjust the size of the spreadsheet. The maximum number of columns has arbitrarily been set to 30. The maximum number of rows has arbitrarily been set to 50. Keep the spreadsheet as small as possible to reduce calculation time.

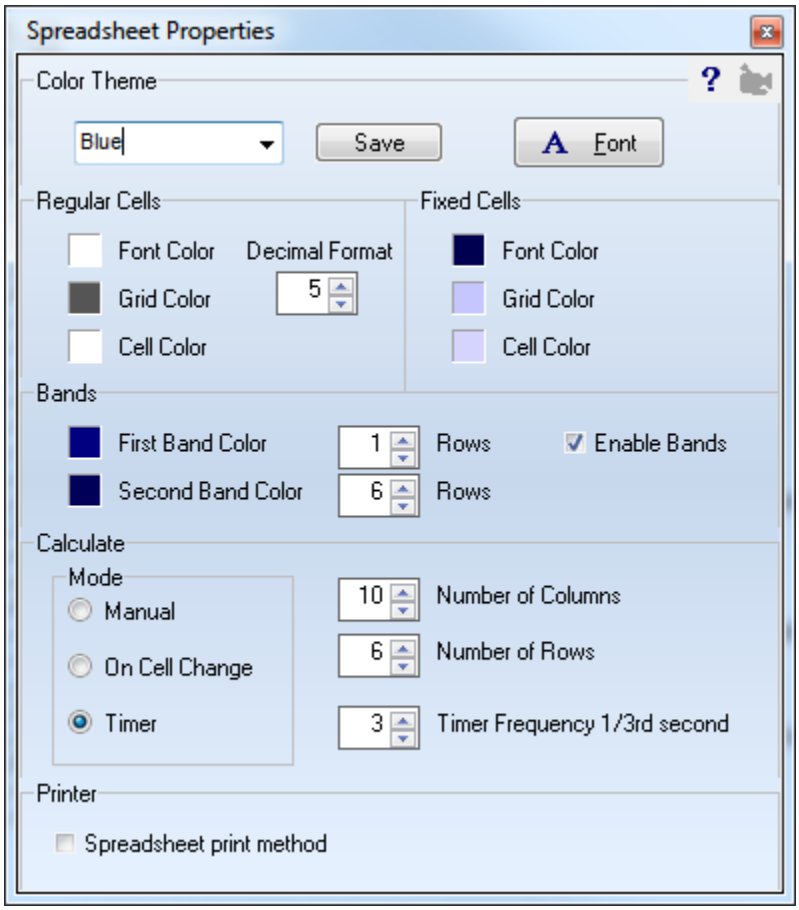
Scoreboard

This example reads bar and study values from 5 charts and displays these values on a spreadsheet.

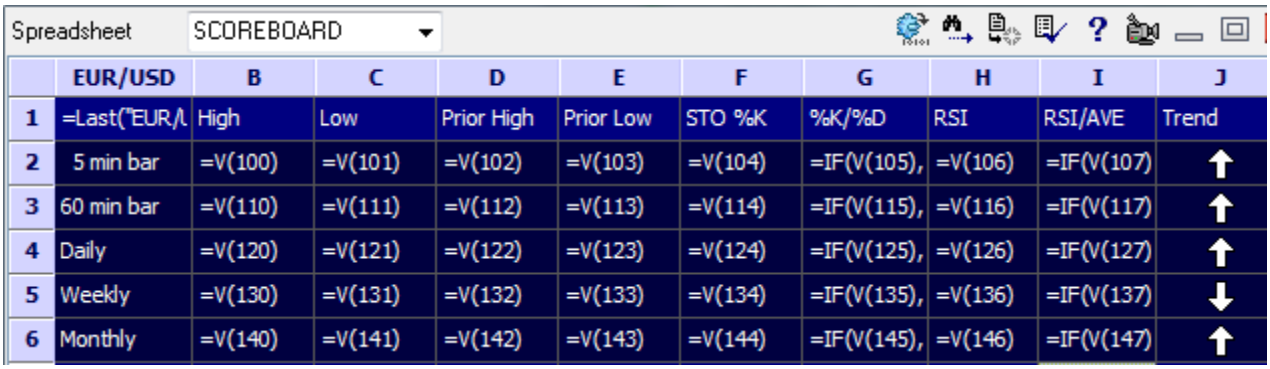
Spreadsheet SCOREBOARD

	EUR/USD	B	C	D	E	F	G	H	I	J
1	1.34444	High	Low	Prior High	Prior Low	STO %K	%K/%D	RSI	RSI/AVE	Trend
2	5 min bar	1.34444	1.34362	1.34490	1.34304	68.15960	Below	64.24729	Below	↓
3	60 min bar	1.34578	1.34304	1.34651	1.34094	80.07141	Above	78.24175	Above	↑
4	Daily	1.34651	1.32523	1.33820	1.30880	72.96116	Above	66.61565	Above	↑
5	Weekly	1.34651	1.32523	1.33820	1.28578	38.50649	Above	42.60505	Below	↓
6	Monthly	1.34651	1.28578	1.34969	1.29696	53.97406	Below	50.39206	Above	↑

Here is the property form for the spreadsheet.



The next image shows the cell formula used.



Spreadsheet SCOREBOARD

	EUR/USD	B	C	D	E	F	G	H	I	J
1	=Last("EUR/L	High	Low	Prior High	Prior Low	STO %K	%K/%D	RSI	RSI/AVE	Trend
2	5 min bar	=V(100)	=V(101)	=V(102)	=V(103)	=V(104)	=IF(V(105),	=V(106)	=IF(V(107)	↑
3	60 min bar	=V(110)	=V(111)	=V(112)	=V(113)	=V(114)	=IF(V(115),	=V(116)	=IF(V(117)	↑
4	Daily	=V(120)	=V(121)	=V(122)	=V(123)	=V(124)	=IF(V(125),	=V(126)	=IF(V(127)	↑
5	Weekly	=V(130)	=V(131)	=V(132)	=V(133)	=V(134)	=IF(V(135),	=V(136)	=IF(V(137)	↓
6	Monthly	=V(140)	=V(141)	=V(142)	=V(143)	=V(144)	=IF(V(145),	=V(146)	=IF(V(147)	↑

Most cells are showing Variable values using the =V(index) function. The variables are being set by a DYO present on each of the 5 charts. An example DYO will be shown later.

Cell A1 has the function: =Last("EUR/USD","FX")

The rest of the cells on row 1 and in column A contain text as shown.

Column G and I use the =IF function to display text based on a Boolean value. The formula in those 2 columns are shown next. Note how a function is used in the 1st parameter position.

G	H	I
%K/%D	RSI	RSI/AVE
=IF(V(105),"Above","Below")	=V(106)	=IF(V(107),"Above","Below")
=IF(V(115),"Above","Below")	=V(116)	=IF(V(117),"Above","Below")
=IF(V(125),"Above","Below")	=V(126)	=IF(V(127),"Above","Below")

Each chart contains a Stochastic, a Relative Strength, an Auto Trends study, and a DYO. The DYO on the 5-minute chart uses variables [100] through [108] to populate row 2 in the spreadsheet. The other charts have similar DYOs, where the only thing that differs are the variables being assigned. The 60-minute chart uses variables [110] through [118], etc.

A	Expression	[100] := High	☐		
B	Expression	[101] := Low	☐		
C	Expression	[102] := Prior High	☐		
D	Expression	[103] := Prior Low	☐		
E	Study	[104] := STO.%K value	☐		
F	Study	[105] := STO.%K >= %D	☐		
G	Study	[106] := RSI.RSI value	☐		
H	Study	[107] := RSI.RSI >= Average	☐		
I	Study	[108] := Trend.Swing direction up	☐		
J	Action	if [108] then Cell(10, 2) := "	☐	⇧	
K	Action	if [108] = 0 then Cell(10, 2) := "	☐	⇩	

Row F and H are storing Boolean flags for use in the spreadsheet's =IF functions.

Row I is storing a Boolean flag used by row J and K to place the arrow markers in Column J. The row K test of 'if [108] = 0 then' is testing for the flag to be False.

DYO Statements

```
Find Spreadsheet( Message )
Cell( #3, #4 ) := Format( #2, [#])
Fontsize( #3, #4 ) := #2
Fontbold( #3, #4 ) := #2 oper 0
if #2 oper 0 then Fontcolor( #3, #4 ) := C
if #2 oper 0 then Cell( #3, #4 ) := Messa


if ## then Rowcolor( #2 ) := Color
Label := Cell( #3, #4)
V := Cell( #3, #4 ) oper #2
V := Cell( #3, #4 ) row move #2 cells
V := Cell( #3, #4 ) col move #2 cells
Cell( #3, #4 ) := Format( V, [#]) row of #2
Cell( #3, #4 ) := Format( V, [#]) col of #2
Spreadsheet.Recalculate
Spreadsheet.Clear
Spreadsheet.Size( #3 cols x #4 rows)
```

The DYO Action category has the following statements for use with a spreadsheet.

Find Spreadsheet – The text in the Label field, or Message field if the Label is blank, is the name of the spreadsheet to find. If the spreadsheet is not found, this statement will open a spreadsheet form with the name.

Cell(col, row) := Format(#2, [#]) – This statement will post the Selection #2 value in the cell and format the number. The format is controlled by the [#] number as per the Format Table. This statement will also do a Marker and Cell Color.

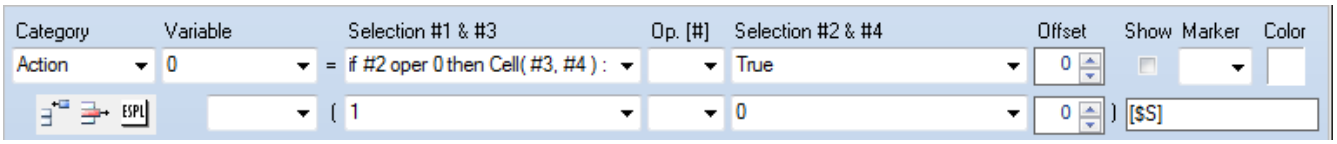
Fontsize(col, row) - Changes the font size for the cell. Size is the Selection #2 value. This statement will do Text, Marker and Cell Color.

Fontbold(col, row) - Changes the font style for the cell. Bold when the expression is True. This statement will do Text, Marker and Cell Color.

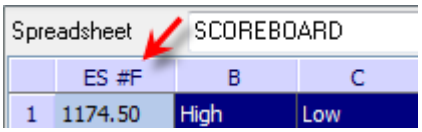
Fontcolor(col, row) - If the #2 oper 0 expression is True, this statement will do Text and Marker, but not Cell Color. The cell font color will be set to the DYO row's color selection. This is an override to the spreadsheet's font color.

Cell(col, row) := Message – If the #2 oper 0 expression is True, this statement will do Text, Marker and Cell Color.

> **Text** - Text is the Label field, or Message field if the Label is blank. This text could be a formula or function. The text may contain reference tags.
>
> **Marker** - If a Marker is selected, the selected Marker is placed to the left of any text. There is no control over the marker color. They will show as seen on the Marker drop down list, where most markers are black.
>
> **Cell Color** - If the Show box is checked, the cell background color is set to the DYO row's color selection. This is an override to the spreadsheet's color or band coloring.



The example writes the chart's symbol in cell(1, 0), which is the 'ES #F' in the Scoreboard example.



Rowcolor(row) - If the ## expression is True, the row's background color is set to the DYO row's color selection. This is an override to the spreadsheet's color or band coloring.

Label := Cell(col, row) - A cell's content is assigned to a DYO Label. The DYO label text can be displayed in the margin of the chart.

V := Cell(col, row) oper #2 - A cell's value is used in the expression with Selection #2 and assigned to the variable V.

V := Cell(col, row) row move #2 cells – Scroll a row of cells. The set size is the #2 value.

V := Cell(col, row) col move #2 cells – Scroll a column of cells. The set size is the #2 value.

Cell(col, row) := Format(V, [#]) row of #2 cells – Format a row of cells. The set size is the #2 value. The format is controlled by the [#] number as per the Format Table.

Cell(col, row) := Format(V, [#]) col of #2 cells – Format a column of cells. The set size is the #2 value. The format is controlled by the [#] number as per the Format Table.

Spreadsheet.Recalculate – The spreadsheet update mode could be on Manual, and the DYO controls when the spreadsheet calculates. Perhaps the DYO is updating Variables or cells on the spreadsheet, then needs the spreadsheet to calculate before spreadsheet results are read by the DYO and used in DYO expressions.

Spreadsheet.Clear – This statement will clear (erase) the contents of the spreadsheet.

Spreadsheet.Size – This statement can be used to change the number of columns and rows.

Format Table

The value posted in the cell is formatted based on the number in the [#] field.

[#]	Description
0..6	Controls the number of decimals shown.
7	Format Time using hh:nn
8	Format Time using hh:nn:ss
9	Format Date using mm:dd:yy
All other values	Use property form for number of decimals

Functions

Single parameter functions

V(index) or GET(index)	returns Variable[index] value, index [0..199]
ABS(parameter)	absolute value
ROUND(parameter)	rounds value
TRUNC(parameter)	truncates value
FRAC(parameter)	returns fractional part of value
FACT(parameter)	factorial of value
INT(parameter)	int part of value
SIN(parameter)	sine of value
COS(parameter)	cosine of value
TAN(parameter)	tangent of value
COTAN(parameter)	cotangent of value
SINH(parameter)	hyperbolic sine of value
COSH(parameter)	hyperbolic cosine of value
TANH(parameter)	hyperbolic tangent of value
COTANH(parameter)	hyperbolic cotangent of value
ASIN(parameter)	arcsin of value
ACOS(parameter)	arccos of value
ATAN(parameter)	arctangent of value
ACOTAN(parameter)	arccotangent of value
LN(parameter)	natural logarithm of value
LOG2(parameter)	base 2 logarithm of value
LOG10(parameter)	base 10 logarithm of value
EXP(parameter)	exponential of value
RAND(parameter)	random between 0 and value
RADIANS(parameter)	converts degrees to radians
DEGREES(parameter)	converts radians to degrees
SQR(parameter)	square of value
SQRT(parameter)	square root of value
CUBE(parameter)	cubic square of value
CHS(parameter)	change sign

Multi parameter functions

SET(index, value)	Assigns Variable[index] := value. Index is in the range [0..199]
CEILING(parameter, significance)	rounds parameter to the nearest multiple of significance
POWER(parameter, exp)	parameter to exponent exp
LT(param1, param2)	larger than : returns 1 if param1>param2 else 0
ST(param1, param2)	smaller than : returns 1 if param1<param2 else 0
EQ(param1, param2)	equal : returns 1 if param1=param2 else 0
CHOOSE(select, param1, param2)	returns param1 when select=0 else param2
ROUNDUP(param, numberdigits)	rounds up a value to the specified number of digits

Cell range functions

SUM(range)	sum of all cell values in range
PRODUCT(range)	product of all cell values in range
AVERAGE(range)	average of all cell values in range
MIN(range)	minimum cell value in range
MAX(range)	maximum cell value in range
COUNT(range)	number of cells in range
COUNTA(range)	number of non blank cells in range
COUNTIF(range, condition)	number of cells meeting condition in range
STDEV(range)	standard deviation of range
STDEVP(range)	standard deviation of total population of range
DEVSQ(range)	sum of squares of deviations of range
VAR(range)	variance of range

Logical functions

AND(parameters)	logical AND function
OR(parameters)	logical OR function
NAND(parameters)	logical NAND function
NOR((parameters)	logical NOR function
XOR((parameters)	logical XOR function
NOT(parameter)	logical NOT function
TRUE	constant returning true
FALSE	constant returning false

Date & Time functions

HOUR(parameter)	gets the hour from a cell containing a valid time string
MIN(parameter)	gets the minute from a cell containing a valid time string
SECOND(parameter)	gets the second from a cell containing a valid time string
DAY(parameter)	gets the day from a cell containing a valid time string
MONTH(parameter)	gets the month from a cell containing a valid time string
YEAR(parameter)	gets the year from a cell containing a valid time string
WEEKDAY(parameter)	gets the day of the week from a valid time string
TODAY	gets the current day
NOW	gets the current time

String functions

LEN(parameter)	returns the length of a string value
LOWER(parameter)	returns string in lowercase
UPPER(parameter)	returns string in uppercase
CONCATENATE(parameter list)	returns concatenated string of parameters
SUBSTITUTE(param text, param oldtext, param new text)	returns string with oldtext replaced by newtext
LEFT(param string, len integer)	returns first len characters of string
RIGHT(param string, len integer)	returns last len characters of string
MID(param string, pos, len integer)	returns len characters starting from position pos in string
TRIM(param)	removes all spaces from text except spaces between words
SEARCH(find text, text)	returns position of string find text in text
LOOKUP(param, range1, range2)	returns the value of the element in range2 that has the index of the matching element in range1 for param
MATCH(param, range)	returns the index of the element param in the range
INDEX(range, val1, val2)	returns the value of element at index val1,val2 in the range
IF(select, param1, param2)	returns param1 if select='1' else returns param2

String parameters entered as strings require the string to be enclosed in double quotes.

Example: =Upper(“abc”)

Constants

PI, E, True, False

Quote functions

LAST(symbol, vendor string)	returns Last price from quote table for symbol
HIGH(symbol, vendor string)	returns High price from quote table for symbol
LOW(symbol, vendor string)	returns Low price from quote table for symbol
OPEN(symbol, vendor string)	returns Open price from quote table for symbol
BID(symbol, vendor string)	returns Bid price from quote table for symbol
ASK(symbol, vendor string)	returns Ask price from quote table for symbol
BIDSIZE(symbol, vendor string)	returns Bid Size from quote table for symbol
ASKSIZE(symbol, vendor string)	returns Ask Size from quote table for symbol
VOLUME(symbol, vendor string)	returns Volume from quote table for symbol
TICKVOL(symbol, vendor string)	returns Tick Volume from quote table for symbol
YESTERDAY(symbol, vendor)	returns Yesterday's Close price from quote table
NET(symbol, vendor string)	returns Net price from quote table for symbol
GETVALUE(symbol, field, vendor)	returns value. Field is one of the quote functions listed here.

String parameters entered as strings require the string to be enclosed in double quotes.

For all Quote functions, the Vendor string is optional. When omitted, the vendor will be looked-up.

Example: =Last(“ES #F”) =Last(“EUR/USD”,”FX”)

=GETVALUE(“EUR/USD”,”Last”,”FX”) {same as =Last(“EUR/USD”,”FX”)}

Vendor String

FX	FXCM forex
IB	Interactive Brokers
ES	eSignal
TA	TransAct Futures
IQ	IQFeed
TB	Trader Bytes
BC	Bar Chart
NJ	Ninja Trader
AT	ATC Brokers
OE	OpenEcry
YH	Yahoo Finance
PB	Playback
EN	Ensign Internet

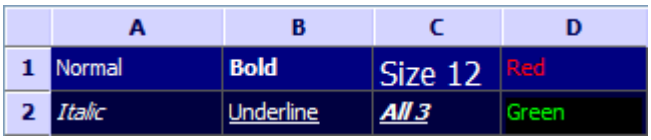
HTML Formatting

The cells in a spreadsheet have support for various HTML tags through which fine control of the display is possible. The supported tags form a subset of the HTML tags.

Supported tags

B : Bold

<B> : start bold text
</B> : end bold text.
Example : This is a <B>test</B>



U : Underline

<U> : start underlined text
</U> : end underlined text
Example : This is a <U>test</U>

I : Italic

<I> : start italic text
</I> : end italic text
Example : This is a <I>test</I>

S : Strikeout

<S> : start strike-through text
</S> : end strike-through text
Example : This is a <S>test</S>

A : Anchor

<A href="value" title="HintValue"> : text after tag is an anchor.
</A> : end of anchor

The 'value' after the href identifier is the anchor. This can be an URL (with ftp,http,mailto,file identifier) or any text. If the value is an URL, the shellexecute function is called.

Examples : This is a <A href="mailto:myemail@mail.com">test</A>
This is a <A href="http://www.ensignsoftware.net">test</A>

Hints for hyperlinks defined in HTML can also be directly be set with the Title attribute. If no Title attribute is specified, the HREF value is used as hint value. Example of a hint in a cell:

A cell hyperlink

FONT : Font Specifier

<FONT face="facevalue" size="sizevalue" color="colorvalue" bgcolor="colorvalue">

Specifies font of text after tag, with

face : name of the font
size : HTML style size if smaller than 5, otherwise pointsize of the font
color : font color with either hexidecimal color specification or Borland style color name, ie. clRed, clYellow, clWhite, clBlack, clBlue, clNavy ... etc.
bgcolor : background color with either hexidecimal color specification or Borland style color name

</FONT> : ends font setting

Examples : This is a <FONT face="Arial" size="12" color="clRed">test</FONT>
This is a <FONT face="Arial" size="12" color="#FF0000">test</FONT>

P : Paragraph

<P align="alignvalue" [bgcolor="colorvalue"]> : starts a new paragraph, with left, right or center alignment. The paragraph background color is set by the optional bgcolor parameter.

</P> : end of paragraph

Example : <P align="right">This is a test</P>
<P align="center">This is a test</P>
<P align="left" bgcolor="#ff0000">This has a red background</P>
<P align="right" bgcolor="clYellow">This has a yellow background</P>

HR : Horizontal Line

<HR> : inserts linebreak with horizontal line
Example: text<HR>

BR : Linebreak

 : inserts a linebreak
Example: 1st line
2nd line

BODY : Body color / background specifier

<BODY bgcolor="colorvalue" background="imagefile specifier"> : sets the background color of the HTML text or the background bitmap file

Example : <BODY bgcolor="clYellow"> : sets background color to yellow
<BODY background="file://c:\test.bmp"> : sets tiled background to file test.bmp

IND : Indent

This is not part of the standard HTML tags but can be used to easily create multi-column text.

<IND x="indent"> : indents with "indent" pixels
Example: This will be <IND x="75">indented 75 pixels.

IMG : Image

<IMG src="specifier:name" [align="specifier"] [width="width"] [height="height"]
[alt="specifier:name"] > : inserts an image at the location.

Specifier can be :

idx : name is the index of the image in the associated imagelist
ssys : name is the index of the small image in the system imagelist or a filename for which the corresponding system imagelist is searched
lsys : same as ssys, but for large system imagelist image
file : name is the full filename specifier
res : name of a resource bitmap (not visible at design time)
no specifier : name of image in an PictureContainer

Optionally, an alignment tag can be included. If no alignment is included, the text alignment with respect to the image is bottom. Other possibilities are: align="top" and align="middle"

The width & height to render the image can be specified as well. If the image is embedded in anchor tags, a different image can be displayed when the mouse is in the image area through the Alt attribute.

Examples : This is an image <IMG src="idx:1" align="top">
This is an image <IMG src="ssys:1">
and another one <IMG src="ssys:worfile.doc">
This is an image <IMG src="file://c:\my documents\test.bmp">
This is an image <IMG src="res://BITMAP1">
This is an image <IMG src="name">

SUB : Subscript

<SUB> : start subscript text
</SUB> : end subscript text
Example: ⁹/₁₆ 

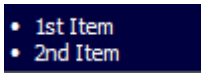
SUP : Superscript

<SUP> : start superscript text
</SUP> : end superscript text

BLINK : Blink

<BLINK> : start blinking text
</BLINK> : stop blinking text
Example : This is <FONT color="clred"><BLINK>blinking red</BLINK></FONT>text.

UL : List



<UL> : start unordered list tag
</UL> : end unordered list
Example : <UL> <LI>List item 1 <LI>List item 2 <UL> <LI> Sub list item A
<LI> Sub list item B </UL> <LI>List item 3 </UL>

LI : List Item

<LI [type="specifier"] [color="color"] [name="imagename"]> : new list item

specifier can be "square" or "circle" or "image" bullet
color sets the color of the square or circle bullet
imagename sets the PictureContainer image name for image to use as bullet

SHAD : Text with Shadow

<SHAD> : start text with shadow
</SHAD> : end text with shadow

Z : Hidden Text

<Z> : start hidden text
</Z> : end hidden text

HI : Hilight

<HI> : start text hilighting
</HI> : stop text hilighting

E : Error Marking

<E> : start error marker
</E> : stop error marker

Special characters

Following standard HTML special characters are supported:

< : less than : <
> : greater than : >
& : &
" : "
  : non breaking space
™ : trademark symbol
€ : euro symbol
§ : section symbol
© : copyright symbol
¶ : paragraph symbol