Excel 4.0 Macro Functions Reference

A COMPREHENSIVE LIST OF MICROSOFT EXCEL 4.0 MACRO FUNCTIONS

Philip Treacy

CO-FOUNDER MY ONLINE TRAINING HUB

CONTENTS

INTRODUCTION

Before Visual Basic for Applications, there were Excel macros, or XLM.

VBA was introduced in Excel version 5.0 so these older XLM macros are also known as Excel 4 Macros.

To use an Excel 4.0 macro does not require any programming. You use them like a function e.g. =GET.CELL(64,A1). This gives you the ColorIndex of the background of cell A1.

However you can't just stick them into your worksheet. If you combine them with defined names they can be very useful and can do things that would normally require a VBA solution.

The definitions for the macro functions in this book are taken from the official Microsoft Excel 4.0 Macros help file.

When I was looking for a reference for these macros I could not easily find one as the old help files are no longer supported in Windows 10.

So I thought I'd put this book together and make it available to anyone that wants it.

Excel 4.0 macros are still working in Excel up to version 2016. That is no guarantee that they will continue to be supported.

A1.R1C1

Displays row and column headings and cell references in either the R1C1 or A1 reference style. A1 is the Microsoft Excel default reference style.

Syntax

A1.R1C1(**logical**)

Logical is a logical value specifying which reference style to use. If logical is TRUE, all worksheets and macro sheets use A1 references; if FALSE, all worksheets and macro sheets use R1C1 references.

Example

The following macro formula displays an alert box asking you to select either A1 or R1C1 reference style. This is useful in an Auto_Open macro if several persons who prefer different reference styles must maintain the same workbook.

```
A1.R1C1(ALERT("Click OK for A1 style; Cancel for R1C1", 1))
```

ABSREF

Returns the absolute reference of the cells that are offset from a reference by a specified amount. You should generally use OFFSET instead of ABSREF. This function is provided for users who prefer to supply an absolute reference in text form.

Syntax

ABSREF(**ref_text**, **reference**)

Ref_text specifies a position relative to reference. Think of ref_text as "directions" from one range of cells to another.

- Ref_text must be an R1C1-style relative reference in the form of text, such as "R[1]C[1]".
- Ref_text is considered relative to the cell in the upper-left corner of reference.

Reference is a cell or range of cells specifying a starting point that ref_text uses to locate another range of cells. Reference can be an external reference.

Remarks

- If you use ABSREF in a function or operation, you will usually get the values contained in the reference instead of the reference itself because the reference is automatically converted to the contents of the reference.
- If you use ABSREF in a function that requires a reference argument, then Microsoft Excel does not convert the reference to a value.
- If you want to work with the actual reference, use the REFTEXT function to convert the active-cell reference to text, which you can then store or manipulate (or convert back to a reference with TEXTREF). See the third example following.

Examples

ABSREF("R[-2]C[-2]", C3) equals $A$1

ABSREF(RELREF(A1, C3), D4) equals $B$2

REFTEXT(ABSREF("R[-2]C[-2]:R[2]C[2]", C3:G7), TRUE) is equivalent to

REFTEXT(ABSREF("R[-2]C[-2]:R[2]C[2]", C3), TRUE), which equals "$A$1:$E$5"

In Microsoft Excel for Windows ABSREF("R[-2]C[-2]", [FINANCE.XLS]Sheet1!C3) equals [FINANCE.XLS]Sheet1!$A$1.

In Microsoft Excel for the Macintosh ABSREF("R[-2]C[-2]", [FINANCE]Sheet1!C3) equals [FINANCE]Sheet1!$A$1

Related Function

RELREF Returns a relative reference

ACTIVATE

Switches to a window if more than one window is open, or a pane of a window if the window is split and its panes are not frozen. Switching to a pane is useful with functions such as VSCROLL, HSCROLL, and GOTO, which operate only on the active pane.

Syntax

ACTIVATE(window_text, pane_num)

ACTIVATE?(window_text, pane_num)

Window_text is text specifying the name of a window to switch to: for example, "Book1" or "Book1:2".

- If a workbook is displayed in more than one window and window_text does not specify which window to switch to, the first window containing that workbook is switched to.
- If window_text is omitted, the active window is not changed.

Pane_num is a number from 1 to 4 specifying which pane to switch to. If pane_num is omitted and the window has more than one pane, the active pane is not changed.

Pane_num	Activates
1	Upper-left pane of the active sheet. If the window is not split, this is the only pane. If the window is split only horizontally, this is the upper pane. If the window is split only vertically, this is the left pane.
2	Upper-right pane of the active sheet. If the window is split only vertically, this is the right pane. If the window is split only horizontally, an error occurs.

3	Lower-left pane of the active sheet. If the window is split only horizontally, this is the lower pane. If the window is split only vertically, an error occurs.
4	Lower-right pane of the active sheet. If the window is split into only two panes either vertically or horizontally, an error occurs.

Related Functions

ACTIVATE.NEXT   Switches to the next window, or switches to the next sheet in a workbook

ACTIVATE.PREV   Switches to the previous window, or switches to the previous sheet in a workbook

DOCUMENTS   Returns the names of the specified open workbooks

FREEZE.PANES   Freezes the panes of a window so that they do not scroll

ON.WINDOW   Runs a macro when you switch to a window

SPLIT   Splits a window

WINDOWS   Returns the names of all open windows

WORKBOOK.SELECT   Select a sheet in a workbook

ACTIVATE.NEXT, ACTIVATE.PREV

Switches to the next or previous window, respectively, or switches to the next or previous sheet in a workbook.

Syntax

ACTIVATE.NEXT(workbook_text)

ACTIVATE.PREV(workbook_text)

Workbook_text   is the name of the workbook for which you want to activate a window.

- If workbook_text is specified, ACTIVATE.NEXT and ACTIVATE.PREV are equivalent to pressing CTRL+PAGE DOWN and CTRL+PAGE UP (in Microsoft Excel for Windows) or COMMAND+PAGE DOWN and COMMAND+PAGE UP (in Microsoft Excel for the Macintosh). These functions switch to the next and previous sheets, respectively.
- If workbook_text is omitted, these functions are equivalent to pressing CTRL+TAB or CTRL+SHIFT+TAB (in Microsoft Excel for Windows) or COMMAND+M or COMMAND+SHIFT+M (in Microsoft Excel for the Macintosh). These functions switch to the next and previous windows, respectively.

Related Functions

ACTIVATE   Switches to a window

ON.WINDOW   Runs a macro when you switch to a window

WORKBOOK.NEXT	Switches to the next sheet in a workbook
WORKBOOK.PREV	Switches to the previous sheet in a workbook
WORKBOOK.SELECT	Select a sheet in a workbook

ACTIVE.CELL

Returns the reference of the active cell in the selection as an external reference.

Syntax

ACTIVE.CELL( )

Remarks

- If an object is selected, ACTIVE.CELL returns the #N/A error value.
- If a chart sheet is active, ACTIVE.CELL returns a zero value.
- If you use ACTIVE.CELL in a function or operation, you will usually get the value contained in the active cell instead of its reference, because the reference is automatically converted to the contents of the reference. See the third example following.
- If you use ACTIVE.CELL in a function that requires a reference argument, then Microsoft Excel does not convert the reference to a value.
- If you want to work with the actual reference, use the REFTEXT XLM function to convert the active-cell reference to text, which you can then store or manipulate (or convert back to a reference with TEXTREF). See the second example following.

Tip Use the following macro formula to verify that the current selection is a cell or range of cells:

```
=ISREF(ACTIVE.CELL( ))
```

Examples

The following macro formula assigns the name Sales to the active cell:

```
SET.NAME("Sales", ACTIVE.CELL())
```

In this example, note that "Sales" refers to a cell on the active worksheet, but the name itself exists only in the macro sheet's list of names. In other words, the preceding formula does not define a name on the worksheet or in the workbook's global list of names.

The following macro formula puts the reference of the active cell into the cell named Temp:

```
FORMULA("="&REFTEXT(ACTIVE.CELL()), Temp)
```

The following macro formula checks the contents of the active cell. If the cell contains only the letter "c" or "s", the macro branches to an area named FinishRefresh:

```
IF(OR(ACTIVE.CELL()="c", ACTIVE.CELL()="s"), GOTO(FinishRefresh))
```

In Microsoft Excel for Windows, if the sheet in the active window is named SALES and A1 is the active cell, then:

`ACTIVE.CELL()` equals SALES!$A$1

In Microsoft Excel for the Macintosh, if the sheet in the active window is named SALES 1 and A1 is the active cell, then:

`ACTIVE.CELL()` equals 'SALES 1'!$A$1

Related Function

SELECT	Selects a cell, worksheet object, or chart item

ACTIVE.CELL.FONT

Equivalent to formatting individual characters in a cell.

Syntax

ACTIVE.CELL.FONT(font, font_style, size, strikethrough, superscript, subscript, outline, shadow, underline, color, normal, background, start_char, char_count)

The arguments for this function are the same as those for FONT.PROPERTIES.

Related Function

FONT.PROPERTIES	Applies a font and other attributes to the selection

ADD.ARROW

Equivalent to clicking the Arrow button on the Drawing toolbar. Adds an arrow to the active chart. If a chart is not the active window, displays an error value.

Syntax

ADD.ARROW( )

Remarks

After you create an arrow with ADD.ARROW, the arrow remains selected, so you can use the arrow form of the PATTERNS function to format the arrow and the FORMAT.MOVE and FORMAT.SIZE functions to change the position and size of the arrow.

Related Functions

CREATE.OBJECT	Creates an object
DELETE.ARROW	Deletes the selected arrow
FORMAT.MOVE	Moves the selected object
FORMAT.SIZE	Changes the size of the selected object
PATTERNS	Changes the appearance of the selected object

ADD.BAR

Creates a new menu bar and returns the bar ID number. Use the bar ID number to identify the menu in functions that display and add menus and commands to the menu bar. You can also use ADD.BAR to restore a built-in menu bar with its original menus and commands.

Syntax

ADD.BAR(bar_num)

Bar_num   is the number of a built-in menu bar that you want to restore. Use ADD.BAR(bar_num) to restore an unaltered version of a built-in menu bar after you have made changes to the menu bar's menus and commands. See ADD.COMMAND for a list of ID numbers for built-in menu bars.

Important   Restoring a built-in menu bar will remove menus and commands added by other macros. Use ADD.COMMAND and ADD.MENU to restore individual commands and menus.

Remarks

- ADD.BAR just creates a new menu bar; it does not display it. Use SHOW.BAR to display a menu bar. The argument to the SHOW.BAR function should be the number returned by ADD.BAR or a reference to the cell containing ADD.BAR.
- You can define up to 15 custom menu bars at one time. If you carry out an ADD.BAR function when more than 15 custom menu bars are already defined, Microsoft Excel returns the #VALUE! error value.

Example

The following formula creates a new menu bar and returns a bar ID number:

```
ADD.BAR()
```

Related Functions

ADD.COMMAND   Adds a command to a menu

ADD.MENU   Adds a menu to a menu bar

DELETE.BAR   Deletes a menu bar

SHOW.BAR   Displays a menu bar

ADD.CHART.AUTOFORMAT

Adds the format of the currently active chart in the current window to the list of custom formats in the Custom Types tab in the Chart Type dialog box.

Syntax

ADD.CHART.AUTOFORMAT(**name_text**, desc_text)

Name_text   is the name you want to appear in the list of custom formats.

Desc_text   is the description you want to appear when the custom format is selected.

Related Function

DELETE.CHART.AUTOFORMAT   Deletes a custom template

ADD.COMMAND

Adds a command to a menu. ADD.COMMAND returns the position number on the menu of the added command. Use ADD.COMMAND to add one or more custom menu commands to a menu on a built-in or custom menu bar. You can also use ADD.COMMAND to restore a deleted built-in command to its original menu.

Syntax

ADD.COMMAND(**bar_num, menu, command_ref**, position1, position2)

Bar_num is the number corresponding to a menu bar or a type of shortcut menu to which you want to add a command.

- Bar_num can be the ID number of a built-in or custom menu bar. The ID number of a custom menu bar is the number returned by the ADD.BAR function.
- Bar_num can also refer to a type of shortcut menu; use menu to identify the specific shortcut menu.

The ID numbers of the built-in menu bars and the types of shortcut menus are listed in the following tables. Short menus are abbreviated versions of the normal Microsoft Excel menus. To turn on short menus, use the SHORT.MENUS function.

Bar_num	**Built-in menu bar**
1	Worksheet and macro sheet (Microsoft Excel 4.0 or later)
2	Chart (Microsoft Excel 4.0 or later)
3	Null (the menu displayed when no workbooks are open)
4	Info
5	Worksheet and macro sheet (short menus, Microsoft Excel 3.0 and earlier)
6	Chart (short menus, Microsoft Excel 3.0 and earlier)
7	Cell, toolbar, and workbook (shortcut menus)
8	Object (shortcut menus)

9	Chart (Microsoft Excel 4.0 or later shortcut menus)
10	Worksheet and macro sheet
11	Chart
12	Visual Basic

Menu is the menu to which you want the new command added.

- Menu can be either the name of a menu as text or the number of a menu.
- If bar_num is 1 through 6, menus are numbered starting with 1 from the left of the menu bar.
- If bar_num is 7, 8, or 9, menu refers to a built-in shortcut menu. The combination of bar_num and menu determines which shortcut menu to modify, as shown in the following table.

Bar_num	**Menu**	**Shortcut menu**
7	1	Toolbars
7	2	Toolbar buttons
7	3	Workbook paging icons in Microsoft Excel 4.0
7	4	Cells (worksheet)
7	5	Column selections
7	6	Row selections
7	7	Workbook tabs
7	8	Cells (macro sheet)

7	9	Workbook title bar
7	10	Desktop (Microsoft Excel for Windows only)
7	11	Module
7	12	Watch pane
7	13	Immediate pane
7	14	Debug code pane
8	1	Drawn or imported objects on worksheets, dialog sheets, and charts
8	2	Buttons on sheets
8	3	Text boxes
8	4	Dialog sheet
9	1	Chart series
9	2	Chart and axis titles
9	3	Chart plot area and walls
9	4	Entire chart
9	5	Chart axes
9	6	Chart gridlines
9	7	Chart floor and arrows

9	8	Chart legend

Note Any commands that you add to the toolbar buttons, watch pane, immediate pane or debug code pane shortcut menus will be dimmed.

Command_ref is an array or a reference to an area on the macro sheet that describes the new command or commands.

- Command_ref must be at least two columns wide. The first column specifies command names; the second specifies macro names. Optional columns can be specified for shortcut keys (in Microsoft Excel for the Macintosh), status bar messages, and custom Help topics, in that order.
- Command_ref is similar to menu_ref in ADD.MENU. For more information about command_ref, see the description of menu_ref in ADD.MENU.
- Command_ref can be the name, as text, of a previously deleted built-in command that you want to restore. You can also use the value returned by the DELETE.COMMAND formula that deleted the command.

Position1 specifies the placement of the new command.

- Use a hyphen (-) to represent a line separating commands on a menu. If you want to place a command before the second separator on a menu, use two hyphens (--), three hyphens for the third separator, and so on.
- Position1 can be a number indicating the position of the command on the menu. Commands are numbered from the top of the menu starting with 1.
- Position1 can be the name of an existing command, as text, above which you want to add the new command.
- If position1 is omitted, the command is added to the bottom of the menu.
- For the toolbar shortcut menu (bar_num 7, menu 1) and the shortcut menu for workbook paging icons in Microsoft Excel version 4.0 (bar_num 7, menu 3), you cannot add commands to the middle of the toolbar name list or the middle of the workbook contents list.

Position2 specifies the placement of the new command on a submenu.

- Position2 can be a number indicating the position of the command on the submenu. Commands are numbered from the top of the menu starting with 1.
- Position2 can be the name of an existing command, as text, above which you want to add the new command.
- If position2 is omitted, the command is added to the main menu, not the submenu.
- To add a command to the bottom of a submenu, use 0 for position2.

Tip In general, use menu and command names rather than numbers for arguments. The numbers assigned to menus and commands change as you add and delete menus and commands. Using names ensures that your menu and command macro functions always refer to the correct items.

Example

The following macro formula adds the command described in cells G16:J16 to the bottom of the worksheet cells shortcut menu:

```
ADD.COMMAND(7, 4, G16:J16)
```

Related Functions

ADD.BAR	Adds a menu bar
ADD.MENU	Adds a menu to a menu bar
ADD.TOOL	Adds one or more buttons to a toolbar
ADD.TOOLBAR	Creates a toolbar with the specified tools
DELETE.COMMAND	Deletes a command from a menu
ENABLE.COMMAND	Enables or disables a menu or custom command
GET.TOOLBAR	Retrieves information about a toolbar
RENAME.COMMAND	Changes the name of a command or menu

ADDIN.MANAGER

Equivalent to clicking the Add-Ins command on the Tools menu. Adds or removes an installed add-in from the working set. The add-in file must already be installed.

Syntax

ADDIN.MANAGER(operation_num, addinname_text, copy_logical)

ADDIN.MANAGER?(operation_num, addinname_text, copy_logical)

Operation_num determines the operation that the add-in manager will perform.

Operation_num	Operation
1	Adds an add-in to the working set, using the descriptive name in the Add-Ins dialog box.
2	Removes an add-in from the working set, using the descriptive name in the Add-Ins dialog box.
3	Adds a new add-in to the list of add-ins that Microsoft Excel knows about. Equivalent to clicking on the Browse button in the Add-Ins dialog box and clicking a file.

Addinname_text specifies the name of the add-in. If operation_num is 1 or 2, use the descriptive name of the add-in, such as "SOLVER". If operation_num is 3, this contains the filename of the add-in.

Copy_logical specifies whether the add-in should be copied to the library directory. This argument is only used if operation_num is 3. If omitted, and the file is on removable media, the user will be asked if they want to copy it to removable media.

ADD.LIST.ITEM

Adds an item in a list box or drop-down control on a worksheet or dialog sheet control.

Syntax

ADD.LIST.ITEM(**text**, index_num)

Text specifies the text of the item to be added. Instead of text, an empty string may be inserted.

Index_num is the list index to be used for the new item. Blank entries are created from the end of the current list to the new item index. If index_num is omitted the new item is appended to the list.

Remarks

If the list box or drop-down box was already filled using the LISTBOX.PROPERTIES function, then adding an item with ADD.LIST.ITEM causes the fillrange contents to be discarded in favor of the new list.

Related Functions

REMOVE.LIST.ITEM	Removes an item in a list box or drop-down box
SELECT.LIST.ITEM	Selects an item in a list box or in a group box

ADD.MENU

Adds a menu to a menu bar. Use ADD.MENU to add a custom menu to a built-in or custom menu bar. You can also use ADD.MENU to restore built-in menus you have deleted with DELETE.MENU. ADD.MENU returns the position number in the menu bar of the new menu.

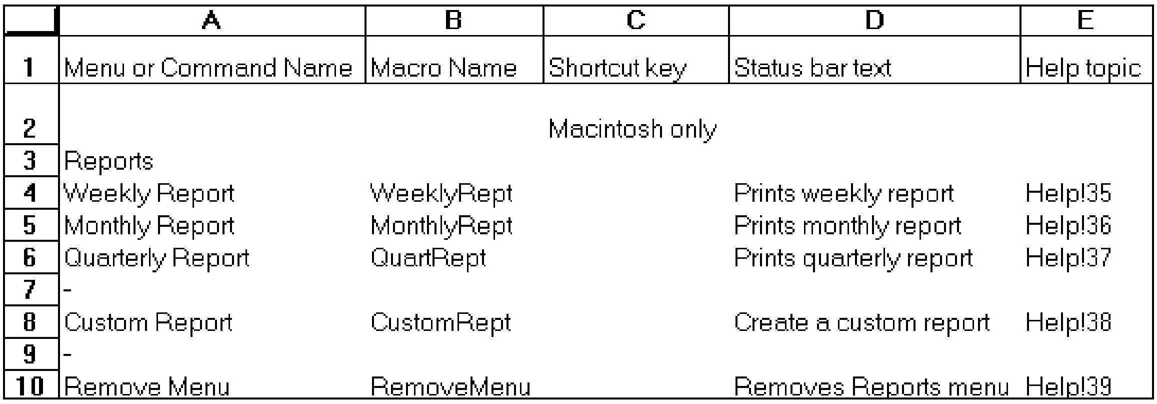
Syntax

ADD.MENU(**bar_num, menu_ref**, position1, position2)

Bar_num is the menu bar to which you want a menu added. Bar_num can be the ID number of a built-in or custom menu bar. See ADD.COMMAND for a list of ID numbers for built-in menu bars.

Menu_ref is an array or a reference to an area on the macro sheet that describes the new menu or the name of a deleted built-in menu you want to restore.

- Menu_ref must be made up of at least two rows and two columns of cells. The upper-left cell of menu_ref specifies the menu title, which is displayed in the menu bar. In the following example, the range A3:E10 is a valid menu_ref.



	A	B	C	D	E
1	Menu or Command Name	Macro Name	Shortcut key	Status bar text	Help topic
2			Macintosh only		
3	Reports				
4	Weekly Report	WeeklyRept		Prints weekly report	Help!35
5	Monthly Report	MonthlyRept		Prints monthly report	Help!36
6	Quarterly Report	QuartRept		Prints quarterly report	Help!37
7	-				
8	Custom Report	CustomRept		Create a custom report	Help!38
9	-				
10	Remove Menu	RemoveMenu		Removes Reports menu	Help!39

The rest of the first column indicates the names of the commands. The corresponding rows in the second column give the names of the macros that run when the commands are chosen.

- You can also specify status-bar text and Help topics in the fourth and fifth columns of menu_ref. In Microsoft Excel for the Macintosh, you can specify shortcut keys in the third column of menu_ref.

Position1 specifies the placement of the new menu. Position can be the name of a menu, as text, or the number of a menu. Menus are numbered from left to right starting with 1. Menus are added to the left of the position specified.

- Use a hyphen (-) to represent a line separating commands on a menu. If you want to place a command before the second separator on a menu, use two hyphens (--), three hyphens for the third separator, and so on.
- If position1 is omitted, the menu is added to the end of the menu bar.
- If there is already a menu at position1, that menu is shifted to the right and the new menu is added in its place.
- If you are using ADD.MENU to restore a deleted built-in menu, you can use the position argument to put it back in its original place on the menu bar. For example, to restore the Data menu on the worksheet and macro sheet menu bar, use position 7. If position1 is omitted, the menu is added to the right of the last menu restored.

Position2 specifies the placement of a submenu.

- Use a hyphen (-) to represent a line separating commands on a menu. If you want to place a command before the second separator on a menu, use two hyphens (--), three hyphens for the third separator, and so on.
- Position2 can be a number indicating the position of the submenu on the menu. Commands are numbered from the top of the menu starting with 1 and include separators.
- Position2 can also be the name, as text, of an existing command above which you want to add the new command.

- If position2 is omitted, the command is added to the main menu, not the submenu.

Example

The following macro formula adds a new menu to the end of the worksheet menu bar, where A10:B15 is the menu_ref describing the menu:

```
ADD.MENU(1, A10:B15)
```

Related Functions

ADD.BAR	Adds a menu bar
ADD.COMMAND	Adds a command to a menu
DELETE.MENU	Deletes a menu
ENABLE.COMMAND	Enables or disables a menu or custom command

ADD.OVERLAY

Equivalent to clicking the Add Overlay command on the Chart menu in Microsoft Excel version 4.0. Adds an overlay to a 2-D chart. If the active chart already has an overlay, ADD.OVERLAY takes no action and returns TRUE. In Microsoft Excel version 5.0 or later, ADD.OVERLAY works with charts that have only one chart type.

Syntax

ADD.OVERLAY( )

Related Functions

ADD.ARROW	Adds an arrow to a chart
LEGEND	Adds a legend to a chart

ADD.TOOL

Adds one or more buttons to a toolbar.

Syntax

ADD.TOOL(**bar_id, position, tool_ref**)

Bar_id is either a number specifying one of the built-in toolbars or the name of a custom toolbar.

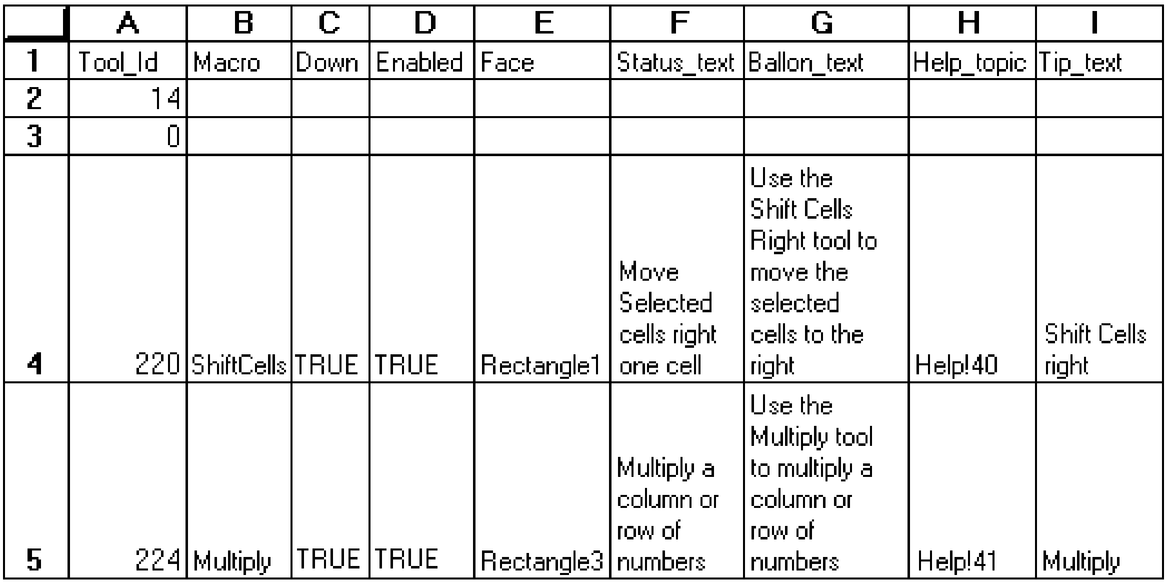
Bar_id	Built-in toolbar
1	Standard

2	Formatting
3	Query and Pivot
4	Chart
5	Drawing
6	TipWizard
7	Forms
8	Stop Recording
9	Visual Basic
10	Auditing
11	WorkGroup
12	Microsoft
13	Full Screen

Position specifies the position of the button within the toolbar. Position starts with 1 at the left side (if horizontal) or at the top (if vertical).

Tool_ref is either a number specifying a built-in button or a reference to an area on the macro sheet that defines a custom button or set of buttons (or an array containing this information).

For customized buttons, the following example shows the components of a button reference area on a macro sheet and defines custom tools. The range A1:I5 is a valid tool_ref. Row 1 refers to a built-in tool. Row 2 defines a gap. For this illustration, values are displayed instead of formulas so that text can wrap in cells.



	A	B	C	D	E	F	G	H	I
1	Tool_Id	Macro	Down	Enabled	Face	Status_text	Ballon_text	Help_topic	Tip_text
2	14								
3	0								
4	220	ShiftCells	TRUE	TRUE	Rectangle1	Move Selected cells right one cell	Use the Shift Cells Right tool to move the selected cells to the right	Help!40	Shift Cells right
5	224	Multiply	TRUE	TRUE	Rectangle3	Multiply a column or row of numbers	Use the Multiply tool to multiply a column or row of numbers	Help!41	Multiply

- Tool_id is a number associated with the tool. A zero specifies a gap on the toolbar. To specify a custom button, use a name, or a number between 201 and 231.

- Macro is the name of, or a quoted R1C1-style reference to, the macro you want to run when the button is clicked.

- Down is a logical value specifying the default image of the tool. If down is TRUE, the button appears depressed into the screen; if FALSE or omitted, it appears normal (up).

- Enabled is a logical value specifying whether the button can be used. If enabled is TRUE, the button is enabled; if FALSE, it is disabled.

- Face specifies a face associated with the tool. Face must be a reference to a picture-type object, for example "Picture 1". If face is omitted, Microsoft Excel uses the default face for the tool.

- Status_text is the text, if any, that you want displayed in the status bar when the button is selected.

- Balloon_text is the balloon help text, if any, associated with the tool. Balloon_text is available only in Microsoft Excel for the Macintosh using system software version 7.0 or later.

- Help_topics is a reference to a topic in a Help file, in the form "filename!topic_number". Help_topics must be text. If help_topics is omitted, HELP displays the Contents topic for Microsoft Excel Help.

- Tip_text is the text, if any, that you want displayed as a ToolTip when the mouse pointer moves over a tool button.

To indicate that a particular component of tool_ref is not used, clear the contents of the corresponding cell.

Remarks

- If you do not want to reserve a section of your macro sheet to define the buttons, you can use an array as the tool_ref argument as shown in the following syntax:

  ADD.TOOL(**bar_id**, position, {**tool_id1**, macro1, down1, enabled1, face1, status_text1, balloon_text1, help_topics1;tool_id2, macro2, down2, enabled2, face2, status_text2, balloon_text2, help_topics2;...})

- Picture objects can be created with the camera button or pasted in from another application. In Microsoft Excel for Windows, the graphic object must be either a Windows bitmap or picture object. In Microsoft Excel for the Macintosh, the object must be a picture object.

Examples

The following macro formula adds a button to Toolbar5. The cell range B6:I6 contains tool_ref.

```
ADD.TOOL("Toolbar5", 6, B6:I6)
```

The following macro formula adds the New Macro Sheet button to the fifth position on the Standard toolbar:

```
ADD.TOOL(1, 5, 6)
```

Related Functions

ADD.COMMAND   Adds a command to a menu

ADD.TOOLBAR   Creates a toolbar with the specified tools

DELETE.TOOL   Deletes a button from a toolbar

DELETE.TOOLBAR   Deletes custom toolbars

RESET.TOOLBAR   Resets a built-in toolbar to its default initial setting

ADD.TOOLBAR

Creates a new toolbar with the specified buttons.

Syntax

ADD.TOOLBAR(**bar_name**, tool_ref)

Bar_name   is a text string identifying the toolbar you want to create.

Tool_ref   is either a number specifying a built-in button or a reference to an area on the macro sheet that defines a custom button or set of buttons (or an array containing this information).

For a complete description of tool_ref, see ADD.TOOL.

Remarks

If you create a toolbar without buttons, use ADD.TOOL to add them. Use SHOW.TOOLBAR to display the toolbar.

Example

The following macro formula creates Toolbar9 with one button in it. The cell range B7:I7 contains tool_ref.

```
ADD.TOOLBAR("Toolbar9", B7:I7)
```

Related Functions

ADD.TOOL   Adds a button to a toolbar

DELETE.TOOL   Deletes a button from a toolbar

DELETE.TOOLBAR   Deletes custom toolbars

RESET.TOOLBAR   Resets a built-in toolbar to its default initial setting

SHOW.TOOLBAR   Hides or displays a toolbar

ALERT

Displays a dialog box and message and waits for you to click a button. Use ALERT instead of MESSAGE if you want to interrupt the flow of a macro and force the user to make a choice or to notice an important message.

Syntax

ALERT(message_text, type_num, help_ref)

Message_text   is the message displayed in the dialog box.

Type_num   is a number from 1 to 3 specifying which type of dialog box to display. If you omit type_num, it is assumed to be 2.

- If type_num is 1, ALERT displays a dialog box containing the OK and Cancel buttons. Click a button to continue or cancel an action. ALERT returns TRUE if you click the OK button and FALSE if you click the Cancel button. See the last example below.
- If type_num is 2 or 3, ALERT displays a dialog box containing an OK button. Click the button to continue, and ALERT returns TRUE. The only difference between specifying 2 or 3 is that ALERT displays a different icon on the left side of the dialog box as shown in the examples below. So, for example, you could use 2 for notes or to present general information, and 3 for errors or warnings.

Help_ref   is a reference to a custom online Help topic, in the form "filename! topic_number".

- If help_ref is present, a Help button appears in the lower-right corner of the alert message. Clicking the Help button starts Help and displays the specified topic.
- If help_ref is omitted, no Help button appears.
- Help_ref must be given in text form.

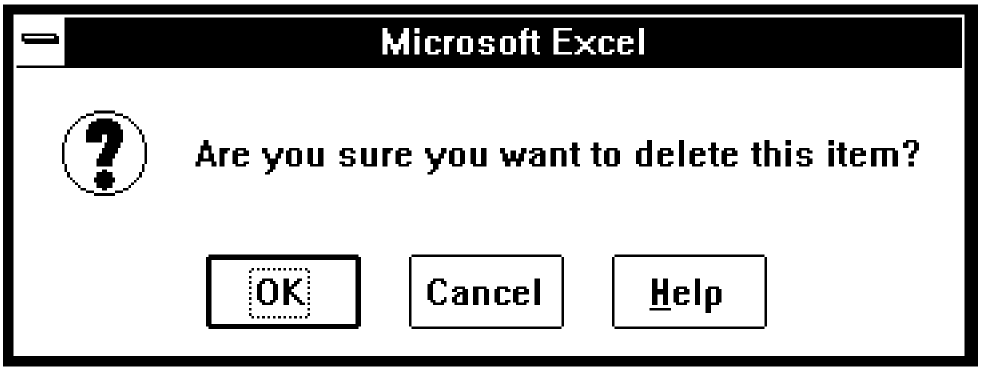
Note   In Microsoft Excel for the Macintosh, the ALERT dialog box is not a movable window.

Examples

The following dialog boxes show the results of using ALERT with type_num 1, 2, and 3. The first and fourth examples include a Help button.

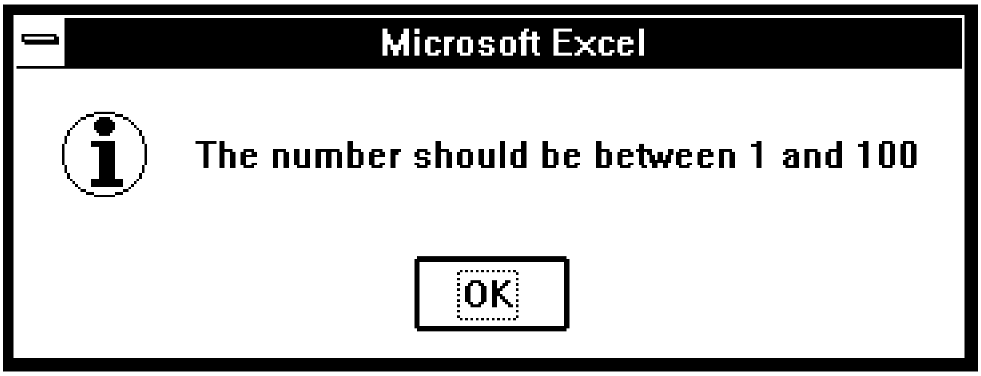
In Microsoft Excel for Windows, the following macro formulas display these three dialog boxes.

```
ALERT("Are you sure you want to delete this item?", 1,
"CUSTHELP.HLP!101")
```



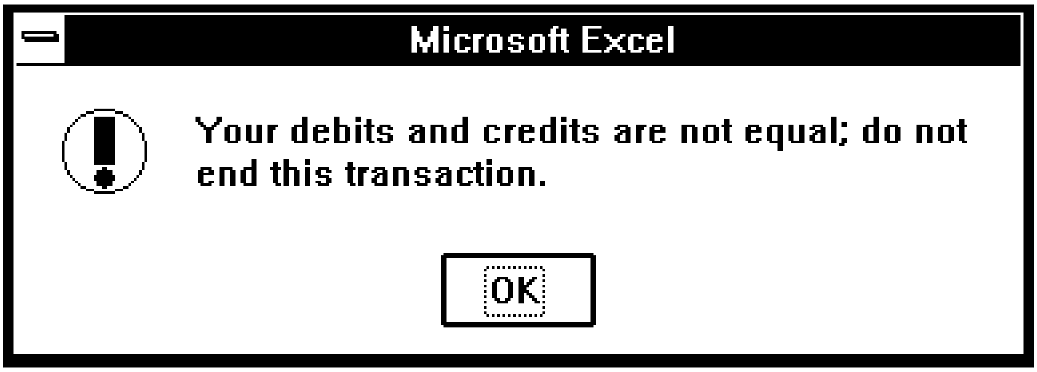
Are you sure you want to delete this item?

```
ALERT("The number should be between 1 and 100", 2)
```



The number should be between 1 and 100

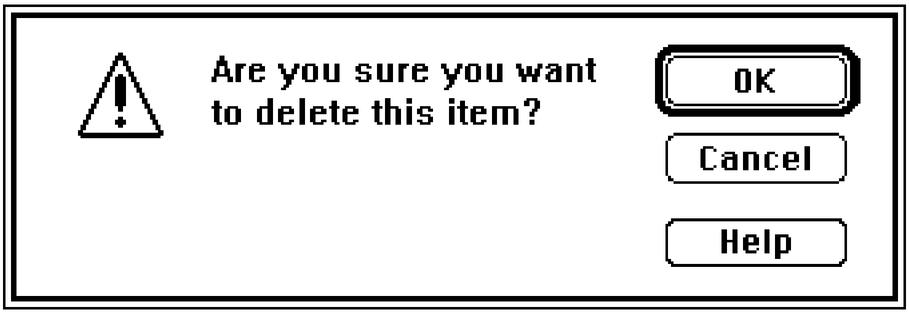
```
ALERT("Your debits and credits are not equal; do not end this
transaction.", 3)
```



Your debits and credits are not equal; do not end this transaction.

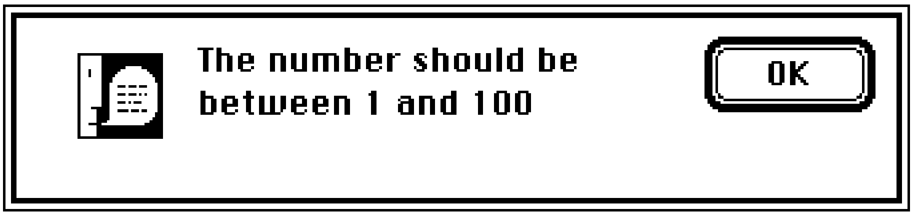
In Microsoft Excel for the Macintosh, the following macro formulas display these three dialog boxes.

```
ALERT("Are you sure you want to delete this item?", 1, "'Custom
Help'!101")
```



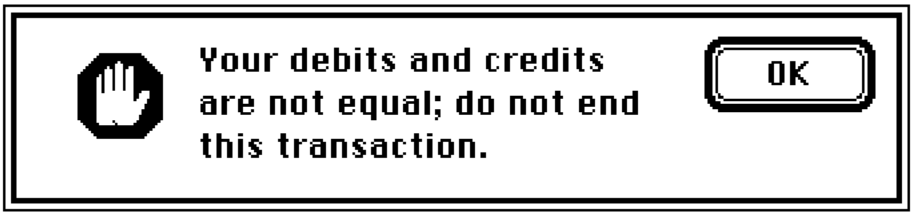
Are you sure you want to delete this item?

```
ALERT("The number should be between 1 and 100", 2)
```



The number should be between 1 and 100

```
ALERT("Your debits and credits are not equal; do not end this
transaction.", 3)
```



Your debits and credits are not equal; do not end this transaction.

A common use of the ALERT function is to give the user a choice of two actions. The following macro formula in an Auto_Open macro asks which reference style to use when the workbook is opened.

```
A1.R1C1(ALERT("Click OK for A1 style; Cancel for R1C1", 1))
```

Related Functions

INPUT	Displays a dialog box for user input
MESSAGE	Displays a message in the status bar

ALIGNMENT

Equivalent to clicking the Alignment tab in the Format Cells dialog box, which is displayed when you click the Cells command on the Format menu. Aligns the contents of the selected cells.

Syntax

ALIGNMENT(horiz_align, wrap, vert_align, orientation, add_indent, shrink_to_fit, merge_cells)

ALIGNMENT?(horiz_align, wrap, vert_align, orientation, add_indent, shrink_to_fit, merge_cells)

Horiz_align is a number from 1 to 7 specifying the type of horizontal alignment, as shown in the following table. If horiz_align is omitted, horizontal alignment does not change.

Horiz_align	Horizontal alignment
1	General
2	Left
3	Center
4	Right
5	Fill
6	Justify
7	Center across selection

Wrap is a logical value corresponding to the Wrap Text check box in the Alignment tab. If wrap is TRUE, Microsoft Excel selects the check box and wraps text in cells; if FALSE, Microsoft Excel clears the check box and does not wrap text. If wrap is omitted, wrapping does not change.

Vert_align is a number from 1 to 4 specifying the vertical alignment of the text. If vert_align is omitted, vertical alignment does not change.

Vert_align	Vertical alignment
1	Top
2	Center
3	Bottom
4	Justify

Orientation is a number from 0 to 4 specifying the orientation of the text. If orientation is omitted, text orientation does not change.

Orientation	Text orientation
0	Horizontal
1	Vertical
2	Upward
3	Downward
4	Automatic (applies to only chart tick labels)

Add_indent This argument is for only Far East versions of Microsoft Excel.

Shrink_to_fit is a logical value corresponding to the Shrink To Fit check box in the Alignment tab.

Merge_cells is a logical value corresponding to the Merge Cells check box in the Alignment tab. If merge_cells is TRUE, Microsoft Excel selects the check box and merges the selected cells; the merged cell contains the value of the left-most cell that was merged. If FALSE, Microsoft Excel clears the check box and unmerges the selected cells; the left-most cell takes the formula or value of the cell that was unmerged. If merge_cells is omitted, cell mergers do not change.

Related Function

FORMAT.TEXT Formats a worksheet text box or a chart text item

ANOVA1

Performs single-factor analysis of variance, which tests the hypothesis that means from several samples are equal.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

ANOVA1(**inprng**, outrng, grouped, labels, alpha)

ANOVA1?(inprng, outrng, grouped, labels, alpha)

Inprng is the input range.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Grouped   is a text character that indicates whether the data in the input range is organized by row or column.

- If grouped is "C" or omitted, then the data is organized by column.
- If grouped is "R", then the data is organized by row.

Labels   is a logical value that describes where the labels are located in the input range, as shown in the following table:

Labels	Grouped	Labels are in
TRUE	"C"	First row of the input range.
TRUE	"R"	First column of the input range.
FALSE or omitted	(ignored)	No labels. All cells in the input range are data.

Alpha   is the significance level at which to evaluate critical values for the F statistic. If omitted, alpha is 0.05.

Related Functions

ANOVA2   Performs two-factor analysis of variance with replication

ANOVA3   Performs two-factor analysis of variance without replication

ANOVA2

Performs two-factor analysis of variance with replication.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

ANOVA2(**inprng**, outrng, **sample_rows**, alpha)

ANOVA2?(inprng, outrng, sample_rows, alpha)

Inprng   is the input range. The input range should contain labels in the first row and column.

Outrng   is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Sample_rows   is the number of rows in each sample.

Alpha   is the significance level at which to evaluate critical values for the F statistic. If omitted, alpha is 0.05.

Related Functions

ANOVA1   Performs single-factor analysis of variance

ANOVA3   Performs two-factor analysis of variance without replication

ANOVA3

Performs two-factor analysis of variance without replication.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

ANOVA3(**inprng**, outrng, labels, alpha)

ANOVA3?(inprng, outrng, labels, alpha)

Inprng is the input range.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Labels is a logical value.

- If labels is TRUE, then the first row and column of the input range contain labels.
- If labels is FALSE or omitted, all cells in inprng are considered data. Microsoft Excel will then generate the appropriate data labels for the output table.

Alpha is the significance level at which to evaluate critical values for the F statistic. If omitted, alpha is 0.05.

Related Functions

ANOVA1	Performs single-factor analysis of variance
ANOVA2	Performs two-factor analysis of variance with replication

APP.ACTIVATE

Switches to an application. Use APP.ACTIVATE to switch to another application that is already running or that you have started by using EXEC.

Syntax

APP.ACTIVATE(title_text, wait_logical)

Important Microsoft Excel for the Macintosh requires system software version 7.0 or later for this function.

Title_text is the name of an application as displayed in its title bar.

- If title_text is omitted, APP.ACTIVATE switches to Microsoft Excel.
- If title_text is not a currently running application, APP.ACTIVATE returns the #VALUE! error value and interrupts the macro.
- Title_text is not necessarily the name of the application file. Use the text that appears in the title bar of the application, which might include the name of the open workbook and path information.

- In Microsoft Excel for the Macintosh, title_text can also refer to the Process Serial Number (PSN) that is returned by an EXEC function.

Wait_logical is a logical value determining when to switch to the application specified by title_text.

- If wait_logical is TRUE, Microsoft Excel waits to be switched to before switching to the application specified by title_text.
- If wait_logical is FALSE or omitted, Microsoft Excel immediately switches to the application specified by title_text.

Remarks

If you are running an application using Microsoft Excel macros, and you want to switch to a third application without switching to Microsoft Excel first, use FALSE as the wait_logical argument. With FALSE, you can use the application title_text without having to switch to Microsoft Excel first.

Examples

The following macro formula switches to Microsoft Word, which is currently displaying the workbook MONTHRPT.DOC in full screen mode:

```
APP.ACTIVATE("MICROSOFT WORD - MONTHRPT.DOC")
```

In Microsoft Excel for the Macintosh, the following macro formula switches to Microsoft Word:

```
APP.ACTIVATE("MICROSOFT WORD")
```

Tip Use an IF statement with APP.ACTIVATE to run an EXEC function if the application you want to switch to is not yet running.

Related Functions

The first five functions following are only for Microsoft Excel for Windows.

APP.MAXIMIZE	Maximizes the Microsoft Excel application window
APP.MINIMIZE	Minimizes the Microsoft Excel application window
APP.MOVE	Moves the Microsoft Excel application window
APP.RESTORE	Restores the Microsoft Excel application window
APP.SIZE	Changes the size of the Microsoft Excel application window
EXEC	Starts another application

APP.ACTIVATE.MICROSOFT

Activates a Microsoft application. If the application is not already activated, this function will load the application into memory.

Syntax

APP.ACTIVATE.MICROSOFT(**app_id**)

App_id   is the ID number associated with the Microsoft Application.

App_id	**Application**
1	Microsoft Word
2	Microsoft PowerPoint
3	Microsoft Mail
4	Microsoft Access (for Microsoft Windows only)
5	Microsoft Fox Pro
6	Microsoft Project
7	Microsoft Schedule +

Remarks

Returns TRUE if the application is activated successfully. Returns FALSE if the application is not activated successfully.

Related Function

APP.ACTIVATE   Switches to an application.

APPLY.NAMES

Equivalent to clicking the Apply command on the Name submenu on the Insert menu. Replaces definitions with their respective names. If no names are defined in the current selection, APPLY.NAMES returns the #VALUE! error value. Use APPLY.NAMES to replace references and values in formulas with names.

Syntax

APPLY.NAMES(**name_array**, ignore, use_rowcol, omit_col, omit_row, order_num, append_last)

APPLY.NAMES?(name_array, ignore, use_rowcol, omit_col, omit_row, order_num, append_last)

Name_array is the name or names to apply as text elements in an array.

- To give more than one name as the argument, you must use an array. For example:
- APPLY.NAMES({"DataRange", "CriteriaRange"})
- If the names indicated by the argument name_array have already replaced all of the appropriate references or values, the #VALUE! error value is returned.

The next four arguments correspond to check boxes and options in the Apply Names dialog box. Arguments that correspond to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box.

Ignore corresponds to the Ignore Relative/Absolute check box.

Use_rowcol corresponds to the Use Row And Column Names check box. If use_rowcol is FALSE, the next three arguments are ignored.

Omit_col corresponds to the Omit Column Name If Same Column check box.

Omit_row corresponds to the Omit Row Name If Same Row check box.

Order_num determines which range name is listed first when a cell reference is replaced by a row-oriented and a column-oriented range name, as shown in the following table.

Order_num	Order of range names
1	Row Column
2	Column Row

Append_last determines whether the names most recently defined are also replaced.

- If append_last is TRUE, Microsoft Excel replaces the definitions of the names in name_array and also replaces the definitions of the last names defined.
- If append_last is FALSE or omitted, Microsoft Excel replaces the definitions of the names in name_array only.

Related Functions

CREATE.NAMES Creates names automatically from text labels on a sheet

DEFINE.NAME Defines a name in the active workbook

LIST.NAMES Lists names and their associated information

APPLY.STYLE

Equivalent to clicking the Style command on the Format menu, selecting a style, and clicking the OK button. Applies a previously defined style to the current selection.

Syntax

APPLY.STYLE(style_text)

APPLY.STYLE?(style_text)

Style_text is the name, as text, of a previously defined style. If style_text is not defined, APPLY.STYLE returns the #VALUE! error value and interrupts the macro. If style_text is omitted, the Normal style is applied to the selection.

Related Functions

DEFINE.STYLE	Defines a cell style
DELETE.STYLE	Deletes a cell style
MERGE.STYLES	Imports styles from another workbook into the active workbook

APP.MAXIMIZE

Equivalent to clicking the Maximize command on the Control menu for the application window. Maximizes the Microsoft Excel window.

Syntax

APP.MAXIMIZE( )

Note This function is only for Microsoft Excel for Windows. You can use this function in macros created with Microsoft Excel for the Macintosh, but it will return the #N/A error value.

Related Functions

APP.ACTIVATE	Switches to an application
APP.MINIMIZE	Minimizes the Microsoft Excel application window
APP.MOVE	Moves the Microsoft Excel application window
APP.RESTORE	Restores the Microsoft Excel application window
APP.SIZE	Changes the size of the Microsoft Excel application window
FULL.SCREEN	Controls full screen display

APP.MINIMIZE

Equivalent to clicking the Minimize command on the Control menu for the application window. Minimizes the Microsoft Excel window.

Syntax

APP.MINIMIZE( )

Note This function is only for Microsoft Excel for Windows. You can use this function in macros created with Microsoft Excel for the Macintosh, but it will return the #N/A error value.

Related Functions

APP.ACTIVATE	Switches to an application
APP.MAXIMIZE	Maximizes the Microsoft Excel application window
APP.MOVE	Moves the Microsoft Excel application window
APP.RESTORE	Restores the Microsoft Excel application window
APP.SIZE	Changes the size of the Microsoft Excel application window

APP.MOVE

Equivalent to clicking the Move command on the Control menu for the application window. Moves the Microsoft Excel window. In Microsoft Excel for Windows, if the application window is already maximized APP.MOVE returns the #VALUE! error value and interrupts the macro.

Syntax

APP.MOVE(x_num, y_num)

APP.MOVE?(x_num, y_num)

Note This function is only for Microsoft Excel for Windows. You can use this function in macros created with Microsoft Excel for the Macintosh, but it will return the #N/A error value.

X_num specifies the horizontal position of the Microsoft Excel window measured in points from the left edge of your screen to the left side of the Microsoft Excel window.

Y_num specifies the vertical position of the Microsoft Excel window measured in points from the top edge of your screen to the top of the Microsoft Excel window.

Remarks

- APP.MOVE?, the dialog-box form of the function, doesn't display a dialog box. Instead, it is equivalent to pressing ALT + SPACEBAR, M or to dragging the title bar with the mouse. With APP.MOVE?, you can move the window with the keyboard or mouse.

- If you specify x_num and/or y_num in the dialog-box form of the function, the window is moved according to the specified coordinates, and you are left in move mode.

Related Functions

APP.ACTIVATE	Switches to an application
APP.MAXIMIZE	Maximizes the Microsoft Excel application window
APP.MINIMIZE	Minimizes the Microsoft Excel application window
APP.RESTORE	Restores the Microsoft Excel application window
APP.SIZE	Changes the size of the Microsoft Excel application window

APP.RESTORE

Equivalent to clicking the Restore command on the Control menu for the application window. Restores the Microsoft Excel window to its previous size and location.

Syntax

APP.RESTORE( )

Note This function is only for Microsoft Excel for Windows. You can use this function in macros created with Microsoft Excel for the Macintosh, but it will return the #N/A error value.

Related Functions

APP.ACTIVATE	Switches to an application
APP.MAXIMIZE	Maximizes the Microsoft Excel application window
APP.MINIMIZE	Minimizes the Microsoft Excel application window
APP.MOVE	Moves the Microsoft Excel application window
APP.SIZE	Changes the size of the Microsoft Excel application window

APP.SIZE

Equivalent to choosing the Size command from the Control menu for the application window. Changes the size of the Microsoft Excel window.

Syntax

APP.SIZE(x_num, y_num)

APP.SIZE?(x_num, y_num)

Note This function is only for Microsoft Excel for Windows. You can use this function in macros created with Microsoft Excel for the Macintosh, but it will return the #N/A error value.

X_num specifies the width of the Microsoft Excel window in points.

Y_num specifies the height of the Microsoft Excel window in points.

APP.SIZE?, the dialog-box form of the function, doesn't display a dialog box. Instead, it is equivalent to pressing ALT, SPACEBAR, S or to dragging a window border with the mouse. Using APP.SIZE?, you can size the window with the keyboard or mouse. If you specify x_num and/or y_num in the dialog-box form of the function, the window is sized according to the specified coordinates, and you are left in size mode.

Related Functions

APP.ACTIVATE	Switches to an application
APP.MAXIMIZE	Maximizes the Microsoft Excel application window
APP.MINIMIZE	Minimizes the Microsoft Excel application window
APP.MOVE	Moves the Microsoft Excel application window
APP.RESTORE	Restores the Microsoft Excel application window

APP.TITLE

Changes the title of the Microsoft Excel application workspace to the title you specify. The title appears at the top of the application window. Use APP.TITLE to control the application title when you're using Microsoft Excel to create a custom application. This function does not apply to Microsoft Excel for the Macintosh.

Syntax

APP.TITLE(text)

Text is the title you want to assign to the Microsoft Excel application workspace. If text is omitted, it is restored to Microsoft Excel.

Remarks

- The custom application title, followed by the individual workbook title, will appear in the application title bar if the workbook is maximized.
- APP.TITLE does not affect DDE communications. You will still refer to the application as "Excel".

Related Function

WINDOW.TITLE Changes the title of the active window

ARGUMENT

Describes the arguments used in a custom function, which is a type of macro, or in a subroutine. A custom function or subroutine must contain one ARGUMENT function for each argument in the macro itself. There are two forms of the ARGUMENT function. In the first form, only name_text is required; in the second form, only reference is required. Use the first form if you want to store the argument as a name. Use the second form if you want to store the argument in a specific cell or cells.

Syntax 1

For name storage

ARGUMENT(**name_text**, data_type_num)

Syntax 2

For cell storage

ARGUMENT(name_text, data_type_num, **reference**)

Name_text is the name of the argument or of the cells containing the argument. Name_text is required if you omit reference.

Data_type_num is a number that determines what type of values Microsoft Excel accepts for the argument. The following table lists the possible data types.

Data_type_num	Type of value

1	Number
2	Text
4	Logical
8	Reference
16	Error
64	Array

- Data_type_num can be a sum of the preceding different numbers to allow for more than one possible type of data. For example, if data_type_num is 7, which is the sum of 1, 2, and 4, then the value can be a number, text, or logical value.
- Data_type_num is an optional argument. If you omit data_type_num, it is assumed to be 7.
- If the value that is passed to the function macro is not of the type specified by data_type_num, Microsoft Excel first attempts to convert it to the specified type. If the value cannot be converted, the macro returns the #VALUE! error value.

Reference is the cell or cells in which you want to store the argument's value.

- If you specify reference, the value that is passed to ARGUMENT is entered as a constant in the specified cell, and name_text becomes an optional argument because you can refer to the cell with either reference or name_text.
- If you omit reference, name_text is defined on the macro sheet and refers to the value that is passed to ARGUMENT. Once name_text is defined, you can use it in formulas.

Remarks

- Custom functions and subroutines can accept from 1 to 29 arguments.
- If a macro contains an ARGUMENT function and you omit the corresponding argument in the function that starts the macro, the macro uses the #N/A error value as the value of the argument.

Examples

To create a custom function that calculates profit, use the following functions to specify arguments for cost, sales, and sales volume:

```
ARGUMENT("UnitsSold", 1)

ARGUMENT("UnitCost", 1)

ARGUMENT("UnitPrice", 1)
```

Related Functions

RESULT   Specifies the data type a custom function returns

VOLATILE   Makes custom functions recalculate automatically

ARRANGE.ALL

Equivalent to clicking the Arrange command on the Window menu. Rearranges open windows and icons and resizes open windows. Also can be used to synchronize scrolling of windows of the active sheet.

Syntax

ARRANGE.ALL(arrange_num, active_doc, sync_horiz, sync_vert)

ARRANGE.ALL?(arrange_num, active_doc, sync_horiz, sync_vert)

Arrange_num   is a number from 1 to 7 specifying how to arrange the windows.

Arrange_num	**Result**
1 or omitted	Tiled (also used to arrange icons in Microsoft Excel for Windows)
2	Horizontal
3	Vertical
4	None
5	Horizontally arranges and sizes the windows based on the position of the active cell.
6	Vertically arranges and sizes the windows based on the position of the active cell.

7	Arranges windows so that they cascade from the upper left to the bottom right of the application workspace.

If you want to change whether the windows are synchronized for scrolling but not how they are arranged, make sure arrange_num is 4.

Active_doc   is a logical value specifying which windows to arrange. If active_doc is TRUE, Microsoft Excel arranges only windows on the active workbook; if FALSE or omitted, all open windows are arranged.

Sync_horiz   is a logical value corresponding to the Sync Horizontal check box in Microsoft Excel version 4.0.

- If sync_horiz is TRUE, Microsoft Excel selects the check box and synchronizes horizontal scrolling.
- If sync_horiz is FALSE or omitted, Microsoft Excel clears the check box, and windows will not be synchronized when you scroll horizontally.
- This argument is used only when active_doc is TRUE.

Sync_vert   is a logical value corresponding to the Sync Vertical check box in Microsoft Excel version 4.0.

- If sync_vert is TRUE, Microsoft Excel selects the check box and synchronizes vertical scrolling.
- If sync_vert is FALSE or omitted, Microsoft Excel clears the check box, and windows will not be synchronized when you scroll vertically.
- This argument is used only when active_doc is TRUE.

Note   If arguments are omitted in the dialog box form of this function, the default values are the previous settings, if any; otherwise the default values are as described above.

Remarks

- After arranging windows, the top or leftmost window is active.
- In Microsoft Excel for Windows, if all windows are minimized, ARRANGE.ALL ignores its arguments, if any, and arranges the corresponding icons horizontally along the bottom of the workspace.

Tip   You can use synchronized horizontal or vertical scrolling when you need to scroll while viewing macro formulas in one window and corresponding macro values in another window of the same macro sheet.

Related Function

ACTIVATE   Switches to a window

ASSIGN.TO.OBJECT

Assigns a macro to the currently select object.

Syntax

ASSIGN.TO.OBJECT(**macro_ref**)

ASSIGN.TO.OBJECT?(macro_ref)

Macro_ref is the name of, or a reference to, the macro you want to run when the object is clicked. If macro_ref is omitted, Microsoft Excel no longer runs the previously specified macro (ASSIGN.TO.OBJECT is turned off).

Remarks

- If an object is not selected, ASSIGN.TO.OBJECT returns the #VALUE! error value and interrupts the macro.
- To change the macro assigned to an object, select the object and use ASSIGN.TO.OBJECT again, using the reference to the new macro as macro_ref. The previous macro is replaced with the new macro.

Related Functions

CREATE.OBJECT Creates an object

RUN Runs a macro

ASSIGN.TO.TOOL

Assigns a macro to be run when a button is clicked with the mouse.

Syntax

ASSIGN.TO.TOOL(**bar_id, position**, macro_ref)

Bar_id specifies the number or name of a toolbar to which you want to assign a macro. For more information about bar_id, see ADD.TOOL.

Position specifies the position of the button within the toolbar. Position starts with 1 at the left side (if horizontal) or at the top (if vertical).

Macro_ref is the name of, or a reference to, the macro you want to run when the button is clicked. If macro_ref is omitted, Microsoft Excel no longer runs the previously specified macro. After canceling the macro, if the button is a built-in button, Microsoft Excel performs the normal default action when the button is clicked. If the button is a custom button, Microsoft Excel displays the Assign Macro dialog box when the button is clicked.

Related Functions

ADD.TOOL Adds one or more buttons to a toolbar

GET.TOOL Returns information about a button or buttons on a toolbar

ATTACH.TEXT

Attaches text to certain parts of the selected chart. Use ATTACH.TEXT to attach text as a title or as a label for an axis or data point.

Syntax

ATTACH.TEXT(**attach_to_num**, series_num, point_num)

ATTACH.TEXT?(attach_to_num, series_num, point_num)

Attach_to_num specifies which item on a chart to attach text to. Attach_to_num is different for 2-D and 3-D charts. Attach_to_num values for 2-D charts are shown in the following table.

Attach_to_num	**Attaches text to**
1	Chart title
2	Value (y) axis
3	Category (x) axis
4	Series and data point
5	Secondary value (y) axis
6	Secondary category (x) axis

Attach_to_num values for 3-D charts are shown in the following table.

Attach_to_num	**Attaches text to**
1	Chart title
2	Value (z) axis
3	Series (y) axis
4	Category (x) axis

5	Series and data point

Series_num specifies the series number if attach_to_num specifies a series or data point. If attach_to_num specifies a series or data point and series_num is omitted, the macro is interrupted.

Point_num specifies the number of the data point, but only if you specify a series number. Point_num is required if series_num is specified, unless the chart is an area chart.

Remarks

When you record adding an axis title or a chart title, Microsoft Excel records both an ATTACH.TEXT function to attach the text and a FONT.PROPERTIES function to make the text bold.

Example

The following macro functions attach the text "Quarterly Sales" to the x (category) axis of the selected chart:

```
ATTACH.TEXT(3)

FORMULA("Quarterly Sales")
```

Related Functions

DATA.LABEL	Assigns text labels to point on a chart
FORMULA	Enters values into a cell or range or onto a chart

ATTACH.TOOLBARS

Displays the Attach Toolbars dialog box, which you use to attach or associate toolbars with documents. The Attach Toolbars dialog box is available when you click the Customize command (View menu, Toolbars submenu), click the Toolbars tab, and then click the Attach button.

Syntax

ATTACH.TOOLBARS?( )

AUTO.OUTLINE

Equivalent to clicking the Auto Outline command on the Group And Outline submenu of the Data menu. Creates an outline within the selection. If a single cell is selected, creates an outline for the entire sheet.

Syntax

AUTO.OUTLINE( )

Related Functions

CLEAR.OUTLINE	Removes outlining from the current sheet

OUTLINE   Creates an outline and defines settings for automatically creating outlines

AXES

Controls whether the axes on a chart are visible. There are two syntax forms of this function. Syntax 1 is for 2-D charts; syntax 2 is for 3-D charts.

Syntax 1

For 2-D charts

AXES(x_primary, y_primary, x_secondary, y_secondary)

AXES?(x_primary, y_primary, x_secondary, y_secondary)

Syntax 2

For 3-D charts

AXES(x_primary, y_primary, z_primary)

AXES?(x_primary, y_primary, z_primary)

Arguments are logical values corresponding to the check boxes in the Axes dialog box.

- If an argument is TRUE, Microsoft Excel selects the check box and displays the corresponding axis.
- If an argument is FALSE, Microsoft Excel clears the check box and hides the corresponding axis.
- If an argument is omitted, the display of that axis is unchanged.

X_primary   corresponds to the primary category (x) axis.

Y_primary   corresponds to the primary value (y) axis.

Z_primary   corresponds to the value (z) axis on the primary 3-D chart.

X_secondary   corresponds to the secondary category (x) axis on a 2-D chart only.

Y_secondary   corresponds to the secondary value (y) axis on a 2-D chart only.

If a 2-D chart has no secondary axis, only the first two arguments are used.

Related Function

GRIDLINES   Controls whether chart gridlines are visible

BEEP

Sounds a tone. Use BEEP to signal a message, a dialog box, or the end of a macro, or whenever you need to get the user's attention.

Syntax

BEEP(tone_num)

Tone_num   is a number from 1 to 4 specifying the tone to be played.

- On most computers, all numbers produce the same sound, the sound that you hear when an error occurs or when you click outside some dialog boxes.
- If tone_num is omitted, it is assumed to be 1.

Remarks

- With a Macintosh, you can control the volume of the tone by using the Control Panel desk accessory.
- With Microsoft Windows version 3.0 or later, you can turn off the tone by using the Control Panel.

Related Functions

ALERT	Displays a dialog box and a message
MESSAGE	Displays a message in the status bar

BORDER

Equivalent to clicking the Border tab in the Format Cells dialog box, which appears when you click the Cells command on the Format menu. Adds a border to the selected cell or range of cells.

Syntax

BORDER(outline, left, right, top, bottom, shade, outline_color, left_color, right_color, top_color, bottom_color)

BORDER?(outline, left, right, top, bottom, shade, outline_color, left_color, right_color, top_color, bottom_color)

Outline, left, right, top, and bottom are numbers from 0 to 7 corresponding to the line styles in the Border dialog box, as shown in the following table.

Argument	Line type
0	No border
1	Thin line
2	Medium line
3	Dashed line
4	Dotted line

5	Thick line
6	Double line
7	Hairline

Note For compatibility with earlier versions of Microsoft Excel, TRUE and FALSE values for the above arguments create a thin border or no border, respectively.

Shade corresponds to the Shade check box in the Border dialog box of Microsoft Excel version 4.0. This argument is included for compatibility only.

Outline_color, left_color, right_color, top_color, and bottom_color are numbers from 1 to 56 corresponding to the Color box in the Border dialog box. Zero corresponds to automatic color.

BREAK

Interrupts a FOR-NEXT, a FOR.CELL-NEXT, or a WHILE-NEXT loop. If BREAK is encountered within a loop, that loop is terminated and the macro proceeds to the statement following the NEXT statement at the end of the current loop.

Syntax

BREAK( )

Example

Use BREAK to test for conditions not anticipated by the FOR or WHILE statement. For example, use the BREAK nested in an IF statement to exit a WHILE-NEXT loop when a certain value is encountered:

```
=IF(Counter=8, BREAK())
```

Related Functions

FOR Starts a FOR-NEXT loop

FOR.CELL Starts a FOR.CELL-NEXT loop

NEXT Ends a FOR-NEXT, FOR.CELL-NEXT, or WHILE-NEXT loop

WHILE Starts a WHILE-NEXT loop

BRING.TO.FRONT

Puts the selected object or objects on top of all other objects. For example, if some worksheet objects are covering part of an embedded chart, you can select the chart and use BRING.TO.FRONT to display the chart on top of the worksheet objects.

Syntax

BRING.TO.FRONT( )

If the selection is not an object or a group of objects, BRING.TO.FRONT returns the #VALUE! error value.

Related Function

SEND.TO.BACK   Sends selected objects behind other objects

CALCULATE.DOCUMENT

Equivalent to choosing the Calc Sheet button in the Calculation tab on the Options dialog, which appears when you choose the Options command from the Tools menu. Calculates only the active worksheet.

Syntax

CALCULATE.DOCUMENT( )

Remarks

If a chart is active, CALCULATE.DOCUMENT returns the #VALUE! error value.

Related Functions

CALCULATE.NOW	Calculates all open workbooks immediately
CALCULATION	Controls calculation settings

CALCULATE.NOW

Equivalent to clicking the Calculation tab in the Options dialog box and then clicking the Calc Now button. Calculates all sheets in all open workbooks. Use CALCULATE.NOW to calculate all open workbooks when calculation is set to manual.

Syntax

CALCULATE.NOW( )

Related Functions

CALCULATE.DOCUMENT	Calculates the active sheet only
CALCULATION	Controls calculation settings

CALCULATION

Controls when and how formulas in open workbooks are calculated. This function is included for compatibility with Microsoft Excel version 4.0. For controlling calculation in Microsoft Excel version 5.0 or later, see OPTIONS.CALCULATION.

Syntax

CALCULATION(**type_num**, iter, max_num, max_change, update, precision, date_1904, calc_save, save_values, alt_exp, alt_form)
CALCULATION?(type_num, iter, max_num, max_change, update, precision, date_1904, calc_save, save_values, alt_exp, alt_form)

Arguments correspond to check boxes and options in the Calculation dialog box. Arguments that correspond to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box.

Type_num   is a number from 1 to 3 indicating the type of calculation.

Type_num	**Type of calculation**
1	Automatic
2	Automatic except tables
3	Manual

Iter   corresponds to the Iteration check box. The default is FALSE.

Max_num   is the maximum number of iterations. The default is 100.

Max_change   is the maximum change of each iteration. The default is 0.001.

Update   corresponds to the Update Remote References check box. The default is TRUE.

Precision   corresponds to the Precision As Displayed check box. The default is FALSE.

Date_1904   corresponds to the 1904 Date System check box. The default is FALSE in Microsoft Excel for Windows and TRUE in Microsoft Excel for the Macintosh.

Calc_save   corresponds to the Recalculate Before Save check box. If calc_save is FALSE, the workbook is not recalculated before saving when in manual calculation mode. The default is TRUE.

Save_values   corresponds to the Save External Link Values check box. The default is TRUE.

Alt_exp   corresponds to the Transition Formula Evaluation check box in the Transition tab of the Options dialog box.

- If alt_exp is TRUE, Microsoft Excel uses a set of rules compatible with that of Lotus 1-2-3 when calculating formulas. Text is treated as 0; TRUE and FALSE are treated as 1 and 0; and certain characters in database criteria ranges are interpreted the same way Lotus 1-2-3 interprets them.
- If alt_exp is FALSE or omitted, Microsoft Excel calculates normally.

Alt_form   corresponds to the Transition Formula Entry check box in the Transition tab of the Options dialog box.

- This argument is available only in Microsoft Excel for Windows.
- If alt_form is TRUE, Microsoft Excel accepts formulas entered in Lotus 1-2-3 style.
- If alt_form is FALSE or omitted, Microsoft Excel only accepts formulas entered in Microsoft Excel style.

Note Microsoft Excel for Windows and Microsoft Excel for the Macintosh use different date systems as their default. Excel for Windows uses the 1900 date system, in which serial numbers correspond to the dates January 1, 1900, through December 31, 9999. Excel for the Macintosh uses the 1904 date system, in which serial numbers correspond to the dates January 1, 1904, through December 31, 9999.

Remarks

Use GET.DOCUMENT to return the current calculation settings for your workbook. For more information, see GET.DOCUMENT.

Related Functions

CALCULATE.DOCUMENT	Calculates the active sheet only
CALCULATE.NOW	Calculates all open workbooks immediately
GET.DOCUMENT	Returns information about a workbook
OPTIONS.CALCULATION	Controls calculation
OPTIONS.TRANSITION	Controls transition options

CALLER

Returns information about the cell, range of cells, command on a menu, tool on a toolbar, or object that called the macro that is currently running. Use CALLER in a subroutine or custom function whose behavior depends on the location, size, name, or other attribute of the caller.

Syntax

CALLER( )

Remarks

- If the custom function is entered in a single cell, CALLER returns the reference of that cell.
- If the custom function was part of an array formula entered in a range of cells, CALLER returns the reference of the range.
- If CALLER appears in a macro called by an Auto_Open, Auto_Close, Auto_Activate, or Auto_Deactivate macro, it returns the name of the calling sheet.
- If CALLER appears in a macro called by a command on a menu, it returns a horizontal array of three elements including the command's position number, the menu number, and the menu bar number.
- If CALLER appears in a macro called by an assigned-to-object macro, it returns the object identifier.
- If CALLER appears in a macro called by a tool on a toolbar, it returns a horizontal array containing the position number and the toolbar name.
- If CALLER appears in a macro called by an ON.DOUBLECLICK or ON.ENTRY function, CALLER returns the name of the chart object identifier or cell reference, if applicable, to which the ON.DOUBLECLICK or ON.ENTRY macro applies.

- If CALLER appears in a macro that was run manually, or for any reason not described above, it returns the #REF! error value.

Examples

If the custom function MACROS!VALUEONE is entered in cell B3 on a sheet named SALES, the nested CALLER function returns the following values.

Nested function	Returns
COLUMN(CALLER())	2
COLUMNS(CALLER())	1
GET.CELL(1, CALLER())	SALES!$B$3
ROW(CALLER())	3
ROWS(CALLER())	1

If the same custom function was entered into an array in cells B2:C3, the following values would be returned.

Nested function	Returns
COLUMN(CALLER())	2
COLUMNS(CALLER())	2
ROW(CALLER())	2
ROWS(CALLER())	2

Related Functions

GET.BAR	Returns the name or position number of menu bars, menus, and commands
GET.CELL	Returns information about the specified cell

CANCEL.COPY

Equivalent to pressing ESC in Microsoft Excel for Windows or ESC or COMMAND+PERIOD in Microsoft Excel for the Macintosh to cancel the marquee after you copy or cut a selection.

Syntax

CANCEL.COPY(render_logical)

Render_logical is a logical value that, if TRUE, places the contents of the Microsoft Excel Clipboard on the Clipboard or, if FALSE or omitted, does not place them on the Clipboard. Render_logical is available only in Microsoft Excel for the Macintosh.

CANCEL.KEY

Disables macro interruption, or specifies a macro to run when a macro is interrupted. Use CANCEL.KEY to control what happens when a macro is interrupted.

Syntax

CANCEL.KEY(**enable**, macro_ref)

Enable specifies whether the macro can be interrupted by pressing ESC in Microsoft Excel for Windows or ESC or COMMAND+PERIOD in Microsoft Excel for the Macintosh.

If enable is	**Then**
FALSE	Pressing ESC or COMMAND+PERIOD does not interrupt a macro
TRUE and macro_ref is omitted	Pressing ESC or COMMAND+PERIOD interrupts a macro
TRUE and macro_ref is specified	Macro_ref runs when ESC or COMMAND+PERIOD is pressed

Macro_ref is a reference to a macro, as a cell reference or a name, that runs when enable is TRUE and ESC or COMMAND+PERIOD is pressed.

Remarks

- CANCEL.KEY affects only the macro that is currently running. Once the macro is stopped by a RETURN or HALT function, ESC or COMMAND+PERIOD is reactivated.
- When CANCEL.KEY is in effect, users can still cancel a dialog box displayed while the macro is running.

Examples

The following macro formula prevents the macro from being interrupted by pressing ESC or COMMAND+PERIOD:

```
CANCEL.KEY(FALSE)
```

The following macro formula reactivates ESC or COMMAND+PERIOD to cancel macro execution:

```
CANCEL.KEY(TRUE)
```

The following line in a macro runs CheckCancel when ESC or COMMAND+PERIOD is pressed:

```
CANCEL.KEY(TRUE, CheckCancel)
```

Related Functions

ERROR	Specifies an action to take if an error occurs while a macro is running
ON.KEY	Runs a macro when a specified key is pressed
ON.TIME	Runs a macro at a specified time

CELL.PROTECTION

Equivalent to choosing the Protection tab in the Format Cells dialog box, which appears when you choose the Cells command from the Format menu. Allows you to control cell protection and display.

Arguments are logical values corresponding to check boxes in the Protection tab. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted and the setting has been previously changed from the defaults, the setting is not changed.

Syntax

CELL.PROTECTION(locked, hidden)

CELL.PROTECTION?(locked, hidden)

Locked corresponds to the Locked check box. The default is TRUE.

Hidden corresponds to the Hidden check box. The default is FALSE.

Remarks

Options selected in the Protection tab of the Format Cells dialog box or with the CELL.PROTECTION function are activated only when the Protect Sheet command is chosen from the Protection submenu on the Tools menu or the PROTECT.DOCUMENT function is used to select protection.

Related Functions

PROTECT.DOCUMENT	Controls protection for the active sheet
SAVE.AS	Saves a workbook and allows you to specify the name, file type, password, backup file, and location of the workbook

CHANGE.LINK

Equivalent to clicking the Change Source button in the Links dialog box, which appears when you click the Links command on the Edit menu. Changes a link from one supporting workbook to another.

Syntax

CHANGE.LINK(**old_text, new_text**, type_of_link)

CHANGE.LINK?(old_text, new_text, type_of_link)

Old_text is the path of the link from the active dependent workbook you want to change.

New_text is the path of the link you want to change to.

Type_of_link is the number 1 or 2 specifying what type of link you want to change.

Type_of_link	Link document type
1 or omitted	Microsoft Excel link
2	DDE/OLE link

Remarks

The workbook whose links you want to change must be active when this function is calculated.

Related Functions

GET.LINK.INFO Returns information about a link

LINKS Returns the name of all linked workbooks

OPEN.LINKS Opens specified supporting workbooks

SET.UPDATE.STATUS Controls the update status of a link

UPDATE.LINK Updates a link to another workbook

CHART.ADD.DATA

Equivalent to dragging data from a worksheet onto a chart. Adds data to an existing chart.

Syntax

CHART.ADD.DATA(**ref**, rowcol, titles, categories, replace, series)

Ref is the cell reference for the data that is being dragged onto the chart

Rowcol is the number 1 or 2 and specifies whether the values corresponding to a particular data series are in rows or columns. Enter 1 for rows or 2 for columns.

Titles is a logical value corresponding to the Series Names In First Column check box (or First Row, depending on the value of rowcol) in the Paste Special dialog box.

- If titles is TRUE, Microsoft Excel selects the check box and uses the contents of the cell in the first column of each row (or first row of each column) as the name of the data series in that row (or column).

- If titles is FALSE, Microsoft Excel clears the check box and uses the contents of the cell in the first column of each row (or first row of each column) as the first data point of the data series.

Categories is a logical value corresponding to the Categories (X Labels) In First Row (or First Column, depending on the value of rowcol) check box in the Paste Special dialog box.

- If categories is TRUE, Microsoft Excel selects the check box and uses the contents of the first row (or column) of the selection as the categories for the chart.

- If categories is FALSE, Microsoft Excel clears the check box and uses the contents of the first row (or column) as the first data series in the chart.

Replace is a logical value corresponding to the Replace Existing Categories check box in the Paste Special dialog box.

- If replace is TRUE, Microsoft Excel selects the check box and applies categories while replacing existing categories with information from the copied cell range.

- If replace is FALSE, Microsoft Excel clears the check box and applies new categories without replacing any old ones.

Series is a number specifying how cells are added to a chart.

Series	**Added as**
1	New series
2	New point(s)

CHART.TREND

A trendline can be added only to these chart types: bar, column, stacked column, scatter, line, and area.

Syntax

CHART.TREND(**type**, ord_per, forecast, backcast, intercept, equation, r_squared, name)

Type is the type of trend or regression.

Number	Type used
1	Linear
2	Logarithmic
3	Polynomial
4	Power
5	Exponential
6	Moving Average

Ord_per depends on type. If type is 3, then ord_per is the order of the polynomial. If type is 6, ord_per is the number of periods for the moving average. If type is neither 3 nor 6, then ord_per is ignored.

Forecast is the number of periods or units to extrapolate the trendline in the positive or forward direction. This argument is ignored for moving averages (type 6). The default is zero.

Backcast is a number specifying the number of periods or units to extrapolate the trendline in the negative or backward direction. This argument is ignored for moving averages (type 6). The default is zero.

Intercept is a number specifying the value of the y-intercept of the trendline, if it is already known. If FALSE or omitted, Microsoft Excel will calculate the y-intercept . This argument is ignored for moving averages.

Equation is a logical value specifying whether the trend equation should be displayed on the chart. If TRUE, the equation will be displayed on the chart. If FALSE or omitted, the equation will not be displayed on the chart.

R_squared is a logical value specifying whether the r-squared equation should be displayed on the chart. If TRUE, the value will be displayed on the chart. If FALSE or omitted, the equation will not be displayed on the chart.

Name is a text string specifying the custom name of the trendline. Can also be a logical value. If TRUE or omitted, the automatic name will be used instead.

Remarks

- A trendline can not be added to a 3-D chart, a stacked chart, or an 100% chart.
- The linear model calculates the least squares fit for a line represented by the equation y = mx + b, where m is the slope and b is the intercept.
- The logarithmic model calculates the least squares fit through points using the equation y = c*ln(x) + b, where c and b are constants.

- The exponential model calculates the least squares fit through points using the following equation:

  $$y = c * e^{(b*x)}$$

  where c and b are constants.

- The polynomial model calculates the least squares fit through points using the following equation:

  $$y = b + (c_1 * x) + (c_2 * x^2) + (c_3 * x^3) + \ldots (c_6 * x^6)$$

  where b, c1, c2, c3, etc. are constants.

- The power model calculates the least squares fit through points using the following equation:

  $$y = cx^b$$

  where b and c are constants.

Related Function

CHART.WIZARD   Equivalent to clicking the ChartWizard button on the Standard toolbar

CHART.WIZARD

Equivalent to clicking the ChartWizard button on the standard or chart toolbar. Creates a chart. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax

CHART.WIZARD(long, **ref**, gallery_num, type_num, plot_by, categories, ser_titles, legend, title, x_title, y_title, z_title, number_cats, number_titles)

CHART.WIZARD?(long, ref, gallery_num, type_num, plot_by, categories, ser_titles, legend, title, x_title, y_title, z_title, number_cats, number_titles)

Long   is a logical value that determines which type of ChartWizard button CHART.WIZARD is equivalent to.

- If long is TRUE, CHART.WIZARD is equivalent to using the five-step ChartWizard button.
- If long is FALSE or omitted, CHART.WIZARD is equivalent to using the two-step ChartWizard button, and gallery_num, type_num, legend, and the title arguments are ignored.

Ref   is a reference to the range of cells on the active worksheet that contains the source data for the chart, or the object identifier of the chart if it has already been created.

Gallery_num   is a number from 1 to 15 specifying the type of chart you want to create.

Gallery_num	Chart
1	Area
2	Bar
3	Column
4	Line
5	Pie
6	Radar
7	XY (scatter)
8	Combination
9	3-D area
10	3-D bar
11	3-D column
12	3-D line
13	3-D pie
14	3-D surface
15	Doughnut

Type_num is a number identifying a formatting option. The first formatting option in any gallery is 1.

Plot_by is the number 1 or 2 and specifies whether the data for each data series is in rows or columns. 1 specifies rows; 2 specifies columns. If plot_by is omitted, Microsoft Excel uses the appropriate value for the chart you're creating.

Categories is the number 1 or 2 and specifies whether the first row or column contains a list of x-axis labels, or data for the first data series. 1 specifies x-axis labels; 2 specifies the first data series. If categories is omitted, Microsoft Excel uses the appropriate value for the chart you're creating. If number_cats is specified, this argument is ignored.

Ser_titles is the number 1 or 2 and specifies whether the first column or row contains series titles, or data for the first data point in each series. 1 specifies series titles; 2 specifies the first data point. If ser_titles is omitted, Microsoft Excel uses the appropriate value for the chart you're creating. If number_titles is specified, this argument is ignored.

Legend is the number 1 or 2 and specifies whether to include a legend. 1 specifies a legend; 2 specifies no legend. If legend is omitted, Microsoft Excel does not include a legend.

For the following arguments, if an argument is omitted or is empty text (""), no title is specified.

Title is text that you want to use as a chart title.

X_title is text that you want to use as an x-axis title.

Y_title is text that you want to use as a y-axis title.

Z_title is text that you want to use as a z-axis title.

Number_cats specifies the number of rows or columns (depending on the value of plot_by) to use for the category labels in the chart. This argument overrides the categories argument.

Number_titles specifies the number of rows or columns (depending on the value of plot_by) to use for the series labels in the chart. This argument overrides the ser_title argument.

Remarks

If you are using the macro recorder, Microsoft Excel records a CREATE.OBJECT and a COPY function when the chart is created and a CHART.WIZARD function when the chart is formatted. You must precede this function with a CREATE.OBJECT function if you are not using the macro recorder.

Related Function

CREATE.OBJECT Creates an object

CHECKBOX.PROPERTIES

Sets various properties of check box and option box controls on a worksheet or dialog sheet.

Syntax

CHECKBOX.PROPERTIES(value, link, accel_text, 3d_shading, accel_text2)

CHECKBOX.PROPERTIES?(value, link, accel_text, 3d_shading, accel_text2,)

Value is the value of the check box or option button setting that determines whether it is selected or not.

Value	Box or Button Setting
0 or FALSE	Off
1 or TRUE	On
2	Mixed

Link is the cell on the sheet to which the check box or option button value is linked. Whenever one of these two controls is changed, the value of the control is entered into the cell. Similarly, whenever the value in the cell is changed, the setting for the corresponding check box or option button is also changed. To clear the link, set this value to an empty string. For example, entering "TRUE" into a cell linked to a check box will select that check box.

3d_shading is a logical value that specifies whether the check box appears as 3-D. If TRUE, the check box will appear as 3-D. If FALSE or omitted, the check box will not be 3-D. This argument is available for only worksheets.

Accel_text is a text string containing the character to use as the control's accelerator key on a dialog sheet. The character is matched against the text of the control, and the first matching character is underlined. When the user presses ALT+accel_text in Microsoft Excel for Windows or COMMAND+accel_text in Microsoft Excel for the Macintosh, the control is clicked.

Accel_text2 is a text string containing the second accelerator key on a dialog sheet. This argument is for only Far East versions of Microsoft Excel.

Remarks

Only controls on dialog sheets can have accelerator keys. Worksheet controls cannot have accelerator keys.

Related Functions

PUSHBUTTON.PROPERTIES Sets the properties of the push button control

EDITBOX.PROPERTIES Sets the properties of an edit box on a worksheet or dialog sheet

LABEL.PROPERTIES Sets the accelerator property of the label and group box control

LISTBOX.PROPERTIES Sets the properties of a list box and drop-down box controls on a worksheet or dialog sheet

CHECK.COMMAND

Adds or removes a check mark to or from a command name on a menu. A check mark beside a command indicates that the command has been chosen.

Syntax

CHECK.COMMAND(**bar_num**, **menu**, **command**, **check**, position)

Bar_num is the menu bar containing the command. Bar_num can be the ID number of a built-in or custom menu bar.

Menu is the menu containing the command. Menu can be either the name of a menu as text or the number of a menu. Menus are numbered starting with 1 from the left of the screen.

Command is the command you want to check or the submenu containing the command you want to check. Command can be the name of the command as text or the number of the command; the first command on a menu is in position 1.

Check is a logical value corresponding to the check mark. If check is TRUE, Microsoft Excel adds a check mark to the command; if FALSE, Microsoft Excel removes the check mark.

position is the name of a command on a submenu that you want to check.

Remarks

- The check mark doesn't affect execution of the command. Microsoft Excel automatically adds and deletes check marks to some commands, such as the name of the active workbook in the Window menu. If you have assigned a check mark to a built-in command that Microsoft Excel automatically changes in response to the user's actions, the check mark will be added or removed as appropriate, and any check marks you have added or deleted with CHECK.COMMAND will be ignored.
- If you use CHECK.COMMAND with a command on a Microsoft Excel version 4.0 menu bar, the corresponding command on the Microsoft Excel version 5.0 or later menu bar will not be effected.

Example

The following macro formula adds a check mark to the Sales command on the Weekly menu on a custom menu bar created by the ADD.BAR function in a cell named Reports:

```
CHECK.COMMAND(Reports, "Weekly", "Sales", TRUE)
```

Related Functions

ADD.COMMAND	Adds a command to a menu
DELETE.COMMAND	Deletes a command from a menu
ENABLE.COMMAND	Enables or disables a menu or custom command
RENAME.COMMAND	Changes the name of a command or menu

CLEAR

Equivalent to clicking the Clear command on the Edit menu. Clears contents, formats, notes, or all of these from the active worksheet or macro sheet. Clears series or formats from the active chart.

Syntax

CLEAR(type_num)

CLEAR?(type_num)

Type_num is a number from 1 to 4 specifying what to clear. Only values 1, 2, and 3 are valid if the selected item is a chart.

On a worksheet or macro sheet, or if an entire chart is selected, the following occurs.

Type_num	Clears
1	All
2	Formats (if a chart, clears the chart format or clears pictures)
3	Contents (if a chart, clears all data series)
4	Comments (this does not apply to charts)

On a chart, if a single point, an entire data series, error bars, or a trend line is selected, the following occurs.

Type_num	Clears
1	Selected series, error bars, or trend line
2	Format in the selected point, series, error bars, or trend line

If type_num is omitted, the default values are set as shown in the following table.

Active sheet	Type_num
Worksheet	3
Macro sheet	3
Chart (with no selection)	1
Chart (with item selected)	Deletes the selected item

Related Function

EDIT.DELETE   Removes cells from a sheet

CLEAR.OUTLINE

Equivalent to clicking the Clear Outline command on the Group And Outline submenu of the Data menu. Clears the outline within the selection. If a single cell is selected, it clears the outline from the entire sheet.

Syntax

CLEAR.OUTLINE( )

Related Functions

AUTO.OUTLINE Creates an outline

OUTLINE Creates an outline and defines settings for automatically creating outlines

CLEAR.ROUTING.SLIP

Equivalent to the Clear button in the Routing Slip dialog box. Clears the routing slip.

Syntax

CLEAR.ROUTING.SLIP(reset_only_logical)

Reset_only_logical is a logical value that specifies whether the routing slip should be cleared.

- This option is valid only after every recipient on the routing slip has received and forwarded the workbook. Setting reset_only_logical to TRUE in this case is equivalent to the Reset button in the routing slip dialog.
- If some recipients have not received or routed the workbook, reset_only_logical is ignored.
- If reset_only_logical is FALSE or omitted and the workbook has been routed to all recipients, then the routing slip is removed from the workbook. A new slip can be subsequently added using ROUTING.SLIP.

CLOSE

Closes the active window. In Microsoft Excel for Windows, CLOSE is equivalent to clicking the Close command on the Document Control menu. In Microsoft Excel for the Macintosh, CLOSE is equivalent to clicking the close box.

Syntax

CLOSE(save_logical, route_logical)

Save_logical is a logical value that specifies whether to save the file before closing the window.

Save_logical	**Result**
TRUE	Saves the file

FALSE	Does not save the file
Omitted	If you've made changes to the file, displays a dialog box asking if you want to save the file

Route_logical is a logical value that specifies whether to route the file after closing it. This argument is ignored if there is not a routing slip present.

Route_logical	**Result**
TRUE	Routes the file
FALSE	Does not route the file
Omitted	If you've specified recipients for routing, displays a dialog box asking if you want to save the file

Remarks

Users of Microsoft Excel versions earlier than 4.0 should note that if the macro sheet containing the function is the active sheet, CLOSE now closes the workbook.

Note When you use the CLOSE function, Microsoft Excel does not run any Auto_Close macros before closing the workbook.

Related Functions

CLOSE.ALL Closes all windows

FILE.CLOSE Closes the active workbook

SAVE Saves the active workbook

CLOSE.ALL

Equivalent to clicking the Close All command on the File menu. The Close All command appears when you hold down SHIFT while selecting the File menu. Closes all protected and unprotected windows and all hidden windows. If unsaved changes have been made to the workbook in one or more windows, a message is displayed asking if you want to save each workbook.

Syntax

CLOSE.ALL( )

Related Functions

CLOSE Closes the active window

FILE.CLOSE Closes the active workbook

QUIT Ends a Microsoft Excel session

SAVE Saves the active workbook

COLOR.PALETTE

Copies a color palette from an open workbook to the active workbook. Use COLOR.PALETTE to share color palettes between workbooks.

Syntax

COLOR.PALETTE(**file_text**)

COLOR.PALETTE?(file_text)

File_text is the name of a workbook, as a text string, that you want to copy a color palette from. The workbook specified by file_text must be open, or COLOR.PALETTE returns the #VALUE! error value and interrupts the macro. If file_text is empty text (""), then COLOR.PALETTE sets colors to the default values.

Related Function

EDIT.COLOR Defines a color on the color palette

COLUMN.WIDTH

Equivalent to choosing the Width command on the Column submenu of the Format menu. Changes the width of the columns in the specified reference.

Syntax

COLUMN.WIDTH(width_num, reference, standard, type_num, standard_num)

COLUMN.WIDTH?(width_num, reference, standard, type_num, standard_num)

Width_num specifies how wide you want the columns to be in units of one character of the font corresponding to the Normal cell style. Width_num is ignored if standard is TRUE or if type_num is provided.

Reference specifies the columns for which you want to change the width.

- If reference is specified, it must be either an external reference to the active worksheet, such as !$A:$C or !Database, or an R1C1-style reference in the form of text, such as "C1:C3", "C[-4]:C[-2]", or "Database".
- If reference is a relative R1C1-style reference in the form of text, it is assumed to be relative to the active cell.
- If reference is omitted, it is assumed to be the current selection.

Standard_num is a logical value corresponding to the Standard Width command from the Column submenu on the Format menu.

- If standard is TRUE, Microsoft Excel sets the column width to the currently defined standard (default) width and ignores width_num.

- If standard is FALSE or omitted, Microsoft Excel sets the width according to width_num or type_num.

Type_num is a number from 1 to 3 corresponding to the Hide, Unhide, or AutoFit Selection commands, respectively, on the Column submenu of the Format menu.

Type_num	**Action taken**
1	Hides the column selection by setting the column width to 0
2	Unhides the column selection by setting the column width to the value set before the selection was hidden
3	Sets the column selection to a best-fit width, which varies from column to column depending on the length of the longest data string in each column

Standard_num specifies how wide the standard width is, and is measured in points. If standard_num is omitted, the standard width setting remains unchanged.

Remarks

- Changing the value of standard_num changes the width of all columns except those that have been set to a custom value.
- If any of the argument settings conflict, such as when standard is TRUE and type_num is 3, Microsoft Excel uses the type_num argument and ignores any arguments that conflict with type_num.
- If you are recording a macro while using a mouse and you change column widths by dragging the column border, Microsoft Excel records the references of the columns using R1C1-style references in the form of text.

Related Function

ROW.HEIGHT Changes the heights of rows

COMBINATION

Changes the format of the active chart to one of six built-in combination chart types.

Syntax

COMBINATION(**type_num**)

COMBINATION?(type_num)

Type_num is a number corresponding to the combination chart you want.

Type_num	Result
1	Column chart overlaid by a line chart
2	Column chart overlaid by a line chart with an independent y-axis scale
3	Line chart overlaid by a line chart with an independent y-axis scale
4	Area chart overlaid by a column chart
5	Column chart overlaid by a line chart containing three data series (for showing stock volumes related to high, low, and closing prices)
6	Column chart overlaid by a line chart containing four data series (for showing stock volumes related to open, high, low, and closing prices

Related Functions

FORMAT.MAIN	Formats a main chart
FORMAT.OVERLAY	Formats an overlay chart

CONSOLIDATE

Equivalent to clicking the Consolidate command on the Data menu. Consolidates data from multiple ranges on multiple worksheets into a single range on a single worksheet.

Syntax

CONSOLIDATE(source_refs, function_num, top_row, left_col, create_links)

CONSOLIDATE?(source_refs, function_num, top_row, left_col, create_links)

Source_refs are references to areas that contain data to be consolidated on the destination worksheet. Source_refs must be in text form and include the full path of the file and the cell reference or named ranges in the workbook to be consolidated. Source_refs are usually external references and must be given as an array, for example: {"SHEET1!IncomeOne", "SHEET2!IncomeTwo"}.

To add or delete source_refs from an existing consolidation on a worksheet, reuse the CONSOLIDATE function, specifying the new source_refs.

Function_num is a number from 1 to 11 that specifies one of the 11 functions you can use to consolidate data. If function_num is omitted, the SUM function, number 9, is used. The functions and their corresponding numbers are listed in the following table.

Function_num	Function
1	AVERAGE
2	COUNT
3	COUNTA
4	MAX
5	MIN
6	PRODUCT
7	STDEV
8	STDEVP
9	SUM
10	VAR
11	VARP

The following arguments correspond to text boxes and check boxes in the Consolidate dialog box. Arguments that correspond to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box.

Top_row   corresponds to the Top Row check box. The default is FALSE.

Left_col   corresponds to the Left Column check box. The default is FALSE.

If top_row and left_col are both FALSE or omitted, the data is consolidated by position.

Create_links   corresponds to the Create Links To Source Data check box.

Remarks

- If you use the CONSOLIDATE function with no arguments and there is a consolidation on the active worksheet, Microsoft Excel reconsolidates, using the sources, function, and position attributes used to create the existing consolidation.

- If you use the CONSOLIDATE function with no arguments and there is no consolidation on the active worksheet, the function returns the #VALUE! error value.

Related Functions

CHANGE.LINK   Changes supporting workbook links

LINKS   Returns the names of all linked workbooks

OPEN.LINKS   Opens specified supporting workbooks

UPDATE.LINK   Updates a link to another workbook

CONSTRAIN.NUMERIC

Equivalent to pressing the Constrain Numeric button. The Constrain Numeric button can be found in the Insert category on the Commands tab (Customize dialog box). The Customize dialog box appears when you choose the Toolbars command from the View menu and then choose the Customize command. Constrains handwriting recognition to numbers and punctuation only. Use this function in a macro to improve the accuracy of handwriting recognition when the user is entering a series of numbers or formulas.

Note   This function is only available if you are using Microsoft Windows for Pen Computing.

Syntax

CONSTRAIN.NUMERIC(numeric_only)

Numeric_only   is a logical value that turns the numeric constraint on or off. If numeric_only is TRUE, only numbers and digits are recognized; if FALSE, all characters are recognized as usual. if numeric_only is omitted, the numeric constraint is toggled.

Remarks

When the numeric constraint is on, Microsoft Excel recognizes only the following symbols:

0 1 2 3 4 5 6 7 8 9 $ # @ % ( ) - + = { } : < > , ? | .

Tip   Use GET.WORKSPACE(45) to make sure you're running Microsoft Windows for Pen Computing.

COPY

Equivalent to clicking the Copy command on the Edit menu. Copies and pastes data or objects.

Syntax

COPY(from_reference, to_reference)

From_reference   is a reference to the cell or range of cells you want to copy. If from_reference is omitted, it is assumed to be the current selection.

To_reference   is a reference to the cell or range of cells where you want to paste what you have copied.

- To_reference should be a single cell or an enlarged multiple of from_reference. For example, if from_reference is a 2 by 4 rectangle, to_reference can be a 4 by 8 rectangle.

- To_reference can be omitted so that you can subsequently paste using the PASTE, PASTE.LINK, or PASTE.SPECIAL functions.

Related Functions

CUT Cuts or moves data or objects

PASTE Pastes cut or copied data

PASTE.LINK Pastes copied data or objects and establishes a link to the source of the data or object

PASTE.SPECIAL Pastes specific components of copied data

COPY.CHART

Equivalent to choosing the Copy Chart command from the Edit menu in Microsoft Excel for the Macintosh version 1.5 or earlier. This function is included only for macro compatibility. You can copy a chart with the COPY.PICTURE function by omitting the appearance_num argument.

Syntax

COPY.CHART(**size_num**)

Size_num is a number describing how to copy the picture and is only available if the current selection is a chart.

Size_num	**Action**
1 or omitted	Copies the chart in the same size as the window on which it is displayed
2	Copies what you would see if you printed the chart

Related Function

COPY.PICTURE Creates a picture of the current selection for use in another program

COPY.PICTURE

Equivalent to choosing the Copy Picture command from the Edit menu. The Copy Picture command appears if you hold down SHIFT while choosing the Edit menu. It copies a chart or range of cells to the Clipboard as a graphic. Use COPY.PICTURE to create an image of the current selection or chart for use in another program.

Syntax

COPY.PICTURE(appearance_num, size_num, type_num)

COPY.PICTURE?(appearance_num, size_num, type_num)

Remarks

Graphics are created differently on screen and on a printer. Thus, the printed picture may look different from the one on screen.

Appearance_num is a number describing how to copy the picture.

Appearance_num	**Action**
1 or omitted	Copies a picture as closely as possible to the picture displayed on your screen
2	Copies what you would see if you printed the selection

Size_num is a number describing how to copy the picture and is only available if the current selection is a chart.

Size_num	**Action**
1 or omitted	Copies the chart in the same size as the window on which it is displayed
2	Copies what you would see if you printed the chart

Type_num is a number specifying the format of the picture. This argument is available only in Microsoft Excel for Windows.

Type_num	**Format of the picture**
1 or omitted	Picture
2	Bitmap

Related Functions

COPY Copies and pastes data or objects

CUT Cuts or moves data or objects

PASTE Pastes cut or copied data

PASTE.PICTURE.LINK Pastes a linked picture of the currently copied area

PASTE.SPECIAL Pastes specific components of copied data

COPY.TOOL

Copies a button face to the Clipboard.

Syntax

COPY.TOOL(bar_id, position)

Bar_id specifies the number or name of a toolbar from which you want to copy the button face. For detailed information about bar_id, see ADD.TOOL.

Position specifies the position of the button within the toolbar. Position starts with 1 at the left side (if horizontal) or at the top (if vertical).

Related Functions

ADD.TOOL Adds one or more buttons to a toolbar

GET.TOOL Returns information about a button or buttons on a toolbar

PASTE.TOOL Pastes a button face from the Clipboard to a specified position on a toolbar

CREATE.NAMES

Equivalent to clicking the Create command on the Name submenu of the Insert menu. Use CREATE.NAMES to quickly create names from text labels on a sheet.

Arguments are logical values corresponding to check boxes in the Create Names dialog box. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE or omitted, Microsoft Excel clears the check box.

Syntax

CREATE.NAMES(top, left, bottom, right)

CREATE.NAMES?(top, left, bottom, right)

Top corresponds to the Top Row check box.

Left corresponds to the Left Column check box.

Bottom corresponds to the Bottom Row check box.

Right corresponds to the Right Column check box.

Remarks

The cell containing the label text that Microsoft Excel uses to create the names is not included in the resulting named range.

Related Functions

APPLY.NAMES Replaces references and values with their corresponding names

DEFINE.NAME Defines a name on the active sheet or macro sheet

DELETE.NAME Deletes a name

FORMULA.GOTO Selects a named area or reference on any open workbook

CREATE.OBJECT

Draws an object on a sheet or macro sheet and returns a value identifying the object created. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax 1

Lines, rectangles, ovals, arcs, pictures, text boxes, and buttons

CREATE.OBJECT(**obj_type**, **ref1**, x_offset1, y_offset1, **ref2**, x_offset2, y_offset2, text, fill, editable)

Syntax 2

Polygons

CREATE.OBJECT(**obj_type**, **ref1**, x_offset1, y_offset1, **ref2**, x_offset2, y_offset2, **array**, fill)

Syntax 3

Embedded charts

CREATE.OBJECT(**obj_type**, **ref1**, x_offset1, y_offset1, **ref2**, x_offset2, y_offset2, xy_series, fill, gallery_num, type_num, plot_visible)

Obj_type   is a number specifying the type of object to create.

Obj_type	**Object**
1	Line
2	Rectangle
3	Oval
4	Arc
5	Embedded chart
6	Text box
7	Button
8	Picture (created with the camera tool)
9	Closed polygon

10	Open polygon
11	Check box
12	Option button
13	Edit box
14	Label
15	Dialog frame
16	Spinner
17	Scroll bar
18	List box
19	Group box
20	Drop down list box

Ref1 is a reference to the cell from which the upper-left corner of the object is drawn, or from which the upper-left corner of the object's bounding rectangle is defined.

X_offset1 is the horizontal distance from the upper-left corner of ref1 to the upper-left corner of the object or to the upper-left corner of the object's bounding rectangle. X_offset1 is measured in points. A point is 1/72nd of an inch. If x_offset1 is omitted, it is assumed to be 0.

Y_offset1 is the vertical distance from the upper-left corner of ref1 to the upper-left corner of the object or to the upper-left corner of the object's bounding rectangle. Y_offset1 is measured in points. If y_offset1 is omitted, it is assumed to be 0.

Ref2 is a reference to the cell from which the lower-right corner of the object is drawn, or from which the lower-right corner of the object's bounding rectangle is defined.

X_offset2 is the horizontal distance from the upper-left corner of ref2 to the lower-right corner of the object or to the lower-right corner of the object's bounding rectangle. X_offset2 is measured in points. If x_offset2 is omitted, it is assumed to be 0.

Y_offset2 is the vertical distance from the upper-left corner of ref2 to the lower-right corner of the object or to the lower-right corner of the object's bounding rectangle. Y_offset2 is measured in points. If y_offset2 is omitted, it is assumed to be 0.

Text specifies the text that appears in a text box or button. If text is omitted for a button, the button is named "Button n", where n is a number. If obj_type is not 6 or 7, text is ignored.

Fill is a logical value specifying whether the object is filled or transparent. If fill is TRUE, the object is filled; if FALSE, the object is transparent; if omitted, the object is filled with an applicable pattern for the object being created.

Array is an n by 2 array of values, or a reference to a range of cells containing values, that indicate the position of each vertex in a polygon, relative to the upper-left corner of the polygon's bounding rectangle.

- A vertex is a point that is defined by a pair of coordinates in one row of array.
- If the polygon contains many vertices, one array may not be sufficient to define it. If the number of characters in the formula exceeds 1024, you must include one or more EXTEND.POLYGON functions. If you're recording a macro, Microsoft Excel automatically records EXTEND.POLYGON functions as needed. For more information, see EXTEND.POLYGON.

Xy_series is a number from 0 to 3 that specifies how data is arranged in a chart and corresponds to options in the Paste Special dialog box.

Xy_series	**Result**
0	Displays a dialog box if the selection is ambiguous
1 or omitted	First row/column is the first data series
2	First row/column contains the category (x) axis labels
3	First row/column contains the x-values; the created chart is an xy (scatter) chart

- Xy_series is ignored unless obj_type is 5 (chart).
- If you want more control over how the data is arranged, use the plot_by, categories, and ser_titles arguments to the CHART.WIZARD function. For more information, see CHART.WIZARD.

Gallery_num is a number from 1 to 15 specifying the type of embedded chart you want to create.

Gallery_num	Chart
1	Area
2	Bar
3	Column
4	Line
5	Pie
6	Radar
7	XY (scatter)
8	Combination
9	3-D area
10	3-D bar
11	3-D column
12	3-D line
13	3-D pie
14	3-D surface
15	Doughnut

Type_num is a number identifying a formatting option for a chart. The formatting options are shown in the dialog box of the AutoFormat command that corresponds to the type of chart you're creating. The first formatting option in any gallery is 1.

Plot_visible   is a logical value that corresponds to the Plot Visible Cells Only checkbox in the Chart tab of the Options dialog box. If FALSE or omitted, all values are plotted.

Editable   is a logical value that determines whether the drop down list box is editable or not. If TRUE, the drop down list box is editable. If FALSE, the drop down list box is not editable. If obj_type is not 20, this argument is ignored.

Remarks

- CREATE.OBJECT returns the object identifier of the object it created. Object identifiers include text describing the object, such as "Text" or "Oval", and a number indicating the order in which the object was created. For example, CREATE.OBJECT returns "Oval 3" after creating an oval that is the third object in the workbook.
- If the offsets are not specified, the object is drawn from the upper-left corner of ref1 to the upper-left corner of ref2.
- If the object is not a picture and either ref1 or ref2 is omitted, CREATE.OBJECT returns the #VALUE! error value and does not create the object.
- CREATE.OBJECT also selects the object.
- You must use the COPY function before the CREATE.OBJECT function to create a chart or a picture.

Tip   To assign a macro to an object, use the ASSIGN.TO.OBJECT function immediately after creating the object.

Related Functions

ASSIGN.TO.OBJECT   Assigns a macro to an object

EXTEND.POLYGON   Adds vertices to a polygon

FORMAT.MOVE   Moves the selected object

FORMAT.SHAPE   Inserts, moves, or deletes vertices of the selected polygon

FORMAT.SIZE   Sizes an object

GET.OBJECT   Returns information about an object

OBJECT.PROPERTIES   Determines an object's relationship to underlying cells

TEXT.BOX   Replaces text in a text box

CREATE.PUBLISHER

Equivalent to clicking the Create Publisher command on the Publishing submenu of the Edit menu. Publishes the selected range or chart to an edition file for use by other Macintosh applications.

Important   This function is only available if you are using Microsoft Excel for the Macintosh with system software version 7.0 or later.

Syntax

CREATE.PUBLISHER(file_text, appearance, size, formats)

CREATE.PUBLISHER?(file_text, appearance, size, formats)

File_text is a text string to be used as the name of the new file that will contain the selected data. If file_text is omitted, Microsoft Excel uses the format "<WorkbookName> Edition #n", where WorkbookName is the name of the workbook from which the publisher is being created, Edition indicates that the file is an edition file, and n is a unique integer.

For example, if you omit file_text and are publishing a selection from a workbook named Seasonal, and it is your third publisher from that workbook in the current work session, the default name of the publisher would be "Seasonal Edition #3".

Appearance specifies whether the selection is to be published as shown on screen or as shown when printed. The default value for appearance is 1 if the selection is a sheet and 2 if the selection is a chart.

Appearance	Selection is published
1	As shown on screen
2	As shown when printed

Size specifies the size at which to publish a chart. Size is only available if a chart is to be published.

Size	Chart is published
1 or omitted	As shown on screen
2	As shown when printed

Formats is number specifying what file format or formats CREATE.PUBLISHER should use when it creates the Edition file.

Formats	File format
1	PICT
2	BIFF
4	RTF
8	VALU

- You can also use the sum of the allowable file formats for formats. For example, a value of 6 specifies BIFF and RTF.
- If formats is omitted and the document is a sheet, formats is assumed to be 15 (all formats); if the document is a chart, formats is assumed to be 1 (PICT).

Related Functions

EDITION.OPTIONS	Sets publisher and subscriber options
GET.LINK.INFO	Returns information about a link
SUBSCRIBE.TO	Inserts contents of an edition into the active workbook
UPDATE.LINK	Updates a link to another workbook

CUSTOMIZE.TOOLBAR

Equivalent to choosing the Toolbars command from the View menu and choosing the Customize button in Microsoft Excel 95. Displays the Customize Toolbars dialog box. In Microsoft Excel 97 or later, this function displays the Commands tab on the Customize dialog box. The Customize dialog box appears when you choose the Toolbars command from the View menu and then choose the Customize command. This function has a dialog-box syntax only.

Syntax

CUSTOMIZE.TOOLBAR?(category)

Category is a number that specifies which category of tools you want displayed in the dialog box. If omitted, the previous setting is used. This argument is for compatibility with Microsoft Excel 95.

Category	Category of tools
1	File
2	Edit
3	Formula
4	Formatting
5	Text Formatting
6	Drawing

7	Macro
8	Charting
9	Utility
10	Data
11	TipWizard
12	Auditing
13	Forms
14	Custom

Related Functions

ADD.TOOLBAR   Creates a new toolbar with the specified tools

SHOW.TOOLBAR   Hides or displays a toolbar

CUSTOM.REPEAT

Allows custom commands to be repeated using the Repeat tool or the Repeat command on the Edit menu. Also allows custom commands to be recorded using the macro recorder.

Syntax

CUSTOM.REPEAT(macro_text, repeat_text, record_text)

Macro_text   is the name of, or a reference to, the macro you want to run when the Repeat command is chosen. If macro_text is omitted, no repeat macro is run, but the custom command can still be recorded.

Repeat_text   is the text you want to use as the repeat command on the Edit menu (for example, "Repeat Reports"). You can omit repeat_text and macro_text if you only want to record the formula specified by record_text when using the macro recorder.

Record_text   is the formula you want to record. For example, if the user clicks a command named Run Reports in Macro 1, the record_text argument would be "=Macro1!RunReports()", where RunReports is the name of the macro called by the Run Reports command.

- References in record_text must be in R1C1 format.

- If record_text is omitted, the macro recorder records normally (a RUN function with the first cell of the macro as its argument).
- If you are not recording a macro, record_text is ignored.

Tip Place CUSTOM.REPEAT at the end of the macro you will want to repeat. If you place it before the end, then the macro formulas that follow CUSTOM.REPEAT may interfere with the desired effects of CUSTOM.REPEAT. The Repeat tool and the Repeat command continue to change as you click subsequent commands that can be repeated.

Example

The following macro formula specifies that the macro RepeatReport on the MenuMacros macro sheet in the current workbook will be run when the Repeat Report command is chosen:

```
CUSTOM.REPEAT("MenuMacros!RepeatReport", "Repeat Report")
```

Related Function

CUSTOM.UNDO   Specifies a macro to run to undo a custom command

CUSTOM.UNDO

Creates a customized Undo tool and Undo or Redo command on the Edit menu for custom commands.

Syntax

CUSTOM.UNDO(**macro_text**, undo_text)

Macro_text   is the name of, or an R1C1-style reference to, the macro you want to run when the Undo command is chosen. Macro_text can be the name or cell reference of a macro.

Undo_text   is the text you want to use as the Undo command.

Example

The following macro function runs the UndoMult macro when the user clicks the Undo Times100 command, a custom command that multiples the current cell by 100.

```
=CUSTOM.UNDO("UndoMult", "&Undo Times100")
```

Tip Use CUSTOM.UNDO directly after the macro functions you want to be able to repeat, because other macro functions following CUSTOM.UNDO might reset the Undo command.

Related Function

CUSTOM.REPEAT   Specifies a macro to run to repeat a custom command

CUT

Equivalent to choosing the Cut command from the Edit menu. Cuts or moves data or objects.

Syntax

CUT(from_reference, to_reference)

From_reference is a reference to the cell or range of cells you want to cut. If from_reference is omitted, it is assumed to be the current selection.

To_reference is a reference to the cell or range of cells where you want to paste what you have cut.

- To_reference should be a single cell or an enlarged multiple of from_reference. For example, if from_reference is a 2 by 4 rectangle, to_reference can be a 4 by 8 rectangle.
- To_reference can be omitted so that you can paste from_reference later using the PASTE or PASTE.SPECIAL functions.

Remarks

The following information may be helpful if you're having problems with CUT updating references in unexpected ways. When you move cells using CUT, formulas that referred to from_reference will refer to to_reference, and formulas that referred to to_reference may return #REF! error values. However, if from_reference or to_reference contains references that are calculated at runtime (for example, `CUT(ACTIVE.CELL(), !B1)`), then Microsoft Excel does not update those references when the CUT function is run, so no error values are returned.

Related Functions

COPY Copies and pastes data or objects

PASTE Pastes cut or copied data

DATA.DELETE

Equivalent to clicking the Delete command on the Data menu in Microsoft Excel version 4.0. Deletes data that matches the current criteria in the current database.

In the dialog-box form, DATA.DELETE?, Microsoft Excel displays a message warning you that matching records will be permanently deleted, and you can approve or cancel. In the plain form, DATA.DELETE, matching records are deleted without any message being displayed.

Syntax

DATA.DELETE( )

DATA.DELETE?( )

DATA.FIND

Equivalent to clicking the Find and Exit Find commands on the Data menu in Microsoft Excel version 4.0. Selects records in the database range which match criteria in the criteria range.

Syntax

DATA.FIND(**logical**)

Logical is a logical value that specifies whether to enter or exit the Data Find mode. If logical is TRUE, Microsoft Excel carries out the Find command; if FALSE, Microsoft Excel carries out the Exit Find command. If logical is omitted, the function toggles between Find and Exit Find.

Related Functions

DATA.FIND.NEXT	Finds next matching record in a database
DATA.FIND.PREV	Finds previous matching record in a database

DATA.FIND.NEXT, DATA.FIND.PREV

Equivalent to pressing the DOWN ARROW or UP ARROW key after the Find command has been chosen from the Data menu in Microsoft Excel version 4.0. Finds the next or previous matching record in a database. If the function cannot find a matching record, it returns the logical value FALSE.

Syntax

DATA.FIND.NEXT( )

DATA.FIND.PREV( )

Related Function

DATA.FIND	Enters or exits Data Find mode

DATA.FORM

Equivalent to clicking the Form command on the Data menu. Displays the data form.

If Microsoft Excel cannot determine what database or list of information to use, the function returns the #VALUE! error value and interrupts the macro.

Syntax

DATA.FORM( )

Remarks

- You can still use custom data forms created in Microsoft Excel version 4.0 or earlier. To edit the definition table of the custom data form, use the Dialog Editor from Microsoft Excel version 4.0.

- The data form can handle up to 32 fields.

DATA.LABEL

Specifies label contents and position.

Syntax

DATA.LABEL(show_option, auto_text, show_key)

Show_option is a number that specifies what type of labels to display.

Show_option	Type displayed
1	none
2	Show value
3	Show percent
4	Show label
5	Show label and percent

Auto_text is a logical value that corresponds the Automatic Checkbox in the Data Labels dialog box. If TRUE, resets a chart's data labels back to their actual values. If FALSE, they are not reset. The Automatic Text checkbox appears only if the label has been selected and its value changed.

Show_key is a logical value that specified whether to show the legend key next to the label. If TRUE, displays the legend key. If FALSE or omitted, does not display the legend key.

DATA.SERIES

Equivalent to clicking the Series command on the Fill submenu of the Edit menu. Use DATA.SERIES to enter an interpolated or incrementally increasing or decreasing series of numbers or dates on a sheet or macro sheet.

Syntax

DATA.SERIES(rowcol, type_num, date_num, step_value, stop_value, trend)

DATA.SERIES?(rowcol, type_num, date_num, step_value, stop_value, trend)

Rowcol is a number that specifies where the series should be entered. If rowcol is omitted, the default value is based on the size and shape of the current selection.

Rowcol	Enter series in
1	Rows
2	Columns

Type_num is a number from 1 to 4 that specifies the type of series.

Type_num	Type of series
1 or omitted	Linear
2	Growth
3	Date
4	AutoFill

Date_num is a number from 1 to 4 that specifies the date unit of the series, as shown in the following table. To use the date_num argument, the type_num argument must be 3.

Date_num	Date unit
1 or omitted	Day
2	Weekday
3	Month
4	Year

Step_value is a number that specifies the step value for the series. If step_value is omitted, it is assumed to be 1.

Stop_value is a number that specifies the stop value for the series. If stop_value is omitted, DATA.SERIES continues filling the series until the end of the selected range.

Trend is a logical value corresponding to the Trend check box. If trend is TRUE, Microsoft Excel generates a linear or exponential trend; if FALSE or omitted, Microsoft Excel generates a standard data series.

Remarks

- If you specify a positive value for stop_value that is lower than the value in the active cell of the selection, DATA.SERIES takes no action.
- If type_num is 4 (AutoFill), Microsoft Excel performs an AutoFill operation just as if you had filled the selection by dragging the fill selection handle or had used the FILL.AUTO macro function.

Related Function

FILL.AUTO Copies cells or automatically fills a selection

DEFINE.NAME

Equivalent to clicking the Define command on the Name submenu of the Insert menu. Defines a name on the active sheet or macro sheet. Use DEFINE.NAME instead of SET.NAME when you want to define a name on the active sheet.

Syntax

DEFINE.NAME(**name_text**, refers_to, macro_type, shortcut_text, hidden, category, local)

DEFINE.NAME?(name_text, refers_to, macro_type, shortcut_text, hidden, category, local)

Name_text is the text you want to use as the name. Names cannot include spaces, and cannot look like cell references.

Refers_to describes what name_text should refer to, and can be any of the following values.

If refers_to is	**Then name_text is**
A number, text, or logical value	Defined to refer to that value
An external reference, such as !$A$1 or SALES!$A$1:$C$3	Defined to refer to those cells
A formula in the form of text, such as "=2*PI()/360" (if the formula contains references, they must be R1C1-style references, such as "=R2C2*(1+RC[-1])")	Defined to refer to that formula
Omitted	Defined to refer to the current selection

The next two arguments, macro_type and shortcut_text, apply only if the sheet in the active window is a macro sheet.

Macro_type   is a number from 1 to 3 that indicates the type of macro.

Macro_type	Type of macro
1	Custom function (also known as a function macro)
2	Command macro.
3 or omitted	None (that is, name_text does not refer to a macro)

Shortcut_text   is a text value that specifies the macro shortcut key. Shortcut_text must be a single letter, such as "z" or "Z".

Hidden   is a logical value specifying whether to define the name as a hidden name. If hidden is TRUE, Microsoft Excel defines the name as a hidden name; if FALSE or omitted, Microsoft Excel defines the name normally.

Category   is a number or text identifying the category of a custom function and corresponds to categories in the Function Category list box.

- Categories are numbered starting with 1, the first category in the list.
- If category is text but is not one of the existing function types, Microsoft Excel creates a new category and assigns your custom function to it.

Local   is a logical value which, if TRUE, defines the name on just the current sheet or macro sheet. If FALSE or omitted, defines the name for all sheets in the workbook.

Remarks

- You can use hidden names to define values that you want to prevent the user from seeing or changing; they do not appear in the Define Name, Paste Name, or Goto dialog boxes. Hidden names can only be created with the DEFINE.NAME macro function.
- If you are recording a macro and you define a name to refer to a formula, Microsoft Excel converts A1-style references to R1C1-style references. For example, if the active cell is C2, and you define the name Previous to refer to =B2, Microsoft Excel records that command as DEFINE.NAME("Previous","=RC[-1]").
- In DEFINE.NAME?, the dialog-box form of the function, if refers_to is not specified, the current selection is proposed in the Refers To box. Also, if a name is not specified, text in the active cell is proposed as the name.

Related Functions

DELETE.NAME   Deletes a name

GET.DEF   Returns a name matching a definition

GET.NAME Returns the definition of a name

NAMES Returns the names defined in a workbook

SET.NAME Defines a name as a value

DEFINE.STYLE

Equivalent to clicking the Define button in the Style dialog box, which appears when you click the Style command on the Format menu. Creates and changes cell styles. There are seven syntax forms of this function. Use syntax 1 of DEFINE.STYLE to define styles based on the format of the active cell. To create a style by specifying number, font, and other formats, use syntaxes 2 through 7 of DEFINE.STYLE.

Syntax 1

Syntaxes 2-7

DEFINE.STYLE SYNTAX 1

Equivalent to clicking the Define button in the Style dialog box, which appears when you click the Style command on the Format menu. Creates and changes cell styles. There are seven syntax forms of this function. Use syntax 1 of DEFINE.STYLE to define styles based on the format of the active cell. To create a style by specifying number, font, and other formats, use syntaxes 2 through 7 of DEFINE.STYLE.

Syntax

DEFINE.STYLE(**style_text**, number, font, alignment, border, pattern, protection)

DEFINE.STYLE?(style_text, number, font, alignment, border, pattern, protection)

Style_text is the name, as text, that you want to assign to the style.

The following arguments are logical values corresponding to check boxes in the Style dialog box. If an argument is TRUE, Microsoft Excel selects the check box and uses the corresponding format of the active cell in the style; if FALSE, Microsoft Excel clears the check box and omits formatting descriptions for that attribute. If style_text is omitted and all selected cells have identical formatting, the default is TRUE; if cells have different formatting, the default is FALSE.

Number corresponds to the Number check box.

Font corresponds to the Font check box.

Alignment corresponds to the Alignment check box.

Border corresponds to the Border check box.

Pattern corresponds to the Pattern check box.

Protection corresponds to the Protection check box.

Related Functions

DEFINE.STYLE Syntaxes 2-7

APPLY.STYLE Applies a style to the selection

DELETE.STYLE Deletes a cell style

MERGE.STYLES Imports styles from another workbook into the active workbook

DEFINE.STYLE SYNTAXES 2 - 7

Equivalent to clicking the Define button in the Style dialog box, which appears when you click the Style command on the Format menu. Creates and changes cell styles. Use one of the following syntax forms of DEFINE.STYLE to select cell formats for a new style or to alter the formats of an existing style. Use syntax 1 of DEFINE.STYLE to define styles based on the format of the active cell.

Syntax 2

Number format, using the arguments from the FORMAT.NUMBER function

DEFINE.STYLE(**style_text, attribute_num**, format_text)

Syntax 3

Font format, using the arguments from the FORMAT.FONT and FONT.PROPERTIES functions

DEFINE.STYLE(**style_text, attribute_num**, name_text, size_num, bold, italic, underline, strike, color, outline, shadow, superscript, subscript)

Syntax 4

Alignment, using the arguments from the ALIGNMENT function

DEFINE.STYLE(**style_text**, **attribute_num**, horiz_align, wrap, vert_align, orientation)

Syntax 5

Border, using the arguments from the BORDER function

DEFINE.STYLE(**style_text**, **attribute_num**, left, right, top, bottom, left_color, right_color, top_color, bottom_color)

Syntax 6

Pattern, using the arguments from the cell form of the PATTERNS function

DEFINE.STYLE(**style_text, attribute_num**, apattern, afore, aback)

Syntax 7

Cell protection, using the arguments from the CELL.PROTECTION function

DEFINE.STYLE(**style_text, attribute_num**, locked, hidden)

Style_text is the name, as text, that you want to assign to the style.

Attribute_num is a number from 2 to 7 that specifies which attribute of the style, such as its font, alignment, or number format, you want to designate with this function.

Attribute_num	**Specifies**
2	Number format
3	Font format

4	Alignment
5	Border
6	Pattern
7	Cell protection

Remarks

- The remaining arguments are different for each form and are identical to arguments in the corresponding function. For example, form 2 of DEFINE.STYLE defines the number format of a style and corresponds to the FORMAT.NUMBER function. The exception is form 5, which does not include every argument for BORDER. For details on the values you can use for these arguments, see the description under the corresponding function.

- If you define a style using one of these forms, then any attributes you don't explicitly define are not changed.

Related Functions

DEFINE.STYLE Syntax 1

ALIGNMENT   Aligns or wraps text in cells

APPLY.STYLE   Applies a style to the selection

BORDER   Adds a border to the selected cell or object

CELL.PROTECTION   Allows you to control cell protection and display

DELETE.STYLE   Deletes a cell style

FONT.PROPERTIES   Applies a font to the selection

FORMAT.NUMBER   Formats numbers, dates, and times in the selected cells

MERGE.STYLES   Imports styles from another workbook into the active workbook

PATTERNS   Changes the appearance of the selected object

DELETE.ARROW

Deletes the selected arrow, either drawn as an arrow with the arrow tool or as a line that is later formatted as an arrow. In Microsoft Excel version 5.0 or later, arrows are named lines.

Syntax

DELETE.ARROW( )

If the selection is not an arrow or a line formatted as an arrow, or if the active window is not a chart, DELETE.ARROW interrupts the macro.

Tip Use the SELECT function (chart syntax), with the number of the arrow (or line) you want to delete in order to select the arrow before using the DELETE.ARROW function. For example, SELECT ("Line 1"). You can also use the CLEAR function to delete the arrow.

Related Functions

CLEAR Clears specified information from the selected cells or chart

DELETE.OVERLAY Deletes the overlay on a chart

DELETE.BAR

Deletes a custom menu bar.

Syntax

DELETE.BAR(**bar_num**)

Bar_num is the ID number of the custom menu bar you want to delete.

Tip Rather than trying to discover the ID number of the menu bar you want to delete, use a reference to the ADD.BAR function that created the bar. For example, the following macro formula deletes the menu bar created by the ADD.BAR function in the cell named ReportsBar:

```
DELETE.BAR(ReportsBar)
```

Related Functions

ADD.BAR Adds a menu bar

SHOW.BAR Displays a menu bar

DELETE.CHART.AUTOFORMAT

Deletes a custom format from the list of formats shown in the Custom Types tab in the Chart Type dialog box.

Syntax

DELETE.CHART.AUTOFORMAT(**name_text**)

Name_text is the template name you want to delete from the list of custom templates.

Related Function

ADD.CHART.AUTOFORMAT Adds a custom template

DELETE.COMMAND

Deletes a command from a custom or built-in menu. Use DELETE.COMMAND to remove commands you don't want the user to have access to or to remove custom commands that you have added.

Syntax

DELETE.COMMAND(**bar_num, menu, command**, subcommand)

Bar_num is the menu bar from which you want to delete the command. Bar_num can be the ID number of a built-in or custom menu bar. See ADD.COMMAND for a list of ID numbers for built-in menu bars and shortcut menus.

Menu is the menu from which you want to delete the command. Menu can be the name of a menu as text or the number of a menu. Menus are numbered starting with 1 from the left of the screen.

Command is the command you want to delete, or the name of a submenu. Command can be the name of the command as text or the number of the command; the first command on a menu is in position 1.

Subcommand is the command you want to delete from a submenu. If you use subcommand, you must use command as the name of the submenu.

Remarks

- If the specified command does not exist, DELETE.COMMAND returns the #VALUE! error value and interrupts the macro.
- After a command is deleted, the command number for all commands below that command is decreased by one.
- When you delete a built-in command, DELETE.COMMAND returns a unique ID number for that command. You can use this ID number with ADD.COMMAND to restore the built-in command to the original menu.

Example

The following macro formula removes the Compile Reports command from the Reports menu on a custom menu bar created by the ADD.BAR function in a cell named Financials.

```
DELETE.COMMAND(Financials, "Reports", "Compile Reports...")
```

Related Functions

ADD.COMMAND	Adds a command to a menu
CHECK.COMMAND	Adds or deletes a check mark to or from a command
ENABLE.COMMAND	Enables or disables a menu or custom command
RENAME.COMMAND	Changes the name of a command or menu

DELETE.FORMAT

Equivalent to deleting the specified format in the Number tab in the Format Cells dialog box, which appears when you click the Cells command on the Format menu, or in the Number tab for selected chart objects. Deletes a specified built-in or custom number format.

Syntax

DELETE.FORMAT(**format_text**)

Format_text is the format given as a text string, for example, "000-00-0000".

Remarks

When you delete a custom number format, all numbers formatted with that number format are formatted with the General format.

Related Functions

FORMAT.NUMBER	Applies a number format to the selection
GET.CELL	Returns information about the specified cell

DELETE.MENU

Deletes a menu or submenu. Use DELETE.MENU to delete menus you have added to menu bars when the supporting macro sheet is closed (using an Auto_Close macro), or any time you want to remove a menu.

Syntax

DELETE.MENU(**bar_num, menu**, submenu)

Bar_num is the menu bar from which you want to delete the menu. Bar_num can be the number of a Microsoft Excel built-in menu bar or the number returned by a previously run ADD.BAR function. For a list of ID numbers for built-in menu bars, see ADD.COMMAND.

Menu is the menu you want to delete. Menu can be either the name of a menu as text or the number of a menu. Menus are numbered starting with 1 from the left of the screen. If the specified menu does not exist, DELETE.MENU returns the #VALUE! error value and interrupts the macro. After a menu is deleted, the menu number for each menu to the right of that menu is decreased by 1.

Submenu is the name of the submenu you want to delete or the number of the menu in the list of commands.

Remarks

You cannot delete a shortcut menu. Instead, use ENABLE.COMMAND to prevent the user from accessing a shortcut menu.

Example

The following macro formula deletes the Reports menu from the custom menu bar created by the ADD.BAR function in a cell named Financials:

```
DELETE.MENU(Financials, "Reports")
```

Related Functions

ADD.MENU	Adds a menu to a menu bar

ADD.BAR   Adds a menu bar

DELETE.BAR   Deletes a menu bar

DELETE.COMMAND   Deletes a command from a menu

ENABLE.COMMAND   Enables or disables a menu or custom command

DELETE.NAME

Equivalent to clicking the Delete button in the Define Name dialog box, which appears when you click the Define command on the Name submenu of the Insert menu. Deletes the specified name.

Syntax

DELETE.NAME(**name_text**)

Name_text   is a text value specifying the name that you want to delete.

Important   Formulas that use names in their arguments may return incorrect or error values when a name used in the formula is deleted.

Related Functions

DEFINE.NAME   Defines a name in the active workbook

GET.NAME   Returns the definition of a name

SET.NAME   Defines a name as a value

DELETE.OVERLAY

Equivalent to clicking the Delete Overlay command on the Chart menu in Microsoft Excel version 4.0. Deletes all overlays from a chart. If the chart has no overlay, DELETE.OVERLAY takes no action and returns TRUE.

Syntax

DELETE.OVERLAY( )

DELETE.STYLE

Equivalent to choosing the Delete button from the Style dialog box, which appears when you choose the Style command from the Format menu. Deletes a style from a workbook. Cells formatted with the deleted style revert to the Normal style.

Syntax

DELETE.STYLE(style_text)

Style_text   is the name of a style to be deleted. If style_text does not exist, DELETE.STYLE returns the #VALUE! error value and interrupts the macro.

Remarks

You can only delete styles from the active workbook. External references are not permitted as part of the style_text argument.

Related Functions

APPLY.STYLE	Applies a style to the selection
DEFINE.STYLE	Creates or changes a cell style
MERGE.STYLES	Merges styles from another workbook into the active workbook

DELETE.TOOL

Equivalent to selecting a button and dragging it to an area other than a toolbar. Deletes a button from a toolbar.

Syntax

DELETE.TOOL(**bar_id, position**)

Bar_id specifies the name or number of a toolbar from which you want to delete a button. For detailed information about bar_id, see ADD.TOOL.

Position specifies the position of the button within the toolbar. Position starts with 1 at the left side (if horizontal) or at the top (if vertical).

Related Functions

ADD.TOOL	Adds one or more buttons to a toolbar
ADD.TOOLBAR	Creates a new toolbar with the specified buttons
DELETE.TOOLBAR	Deletes custom toolbars

DELETE.TOOLBAR

Equivalent to clicking the Delete button in the Toolbars dialog box, which appears when you click the Customize command (View menu, Toolbars submenu). Deletes a custom toolbar.

Syntax

DELETE.TOOLBAR(**bar_name**)

Bar_name specifies the name of the toolbar that you want to delete. For detailed information about bar_name, see ADD.TOOL.

Remarks

- You cannot delete built-in toolbars.
- If DELETE.TOOLBAR successfully deletes the toolbar, it returns TRUE. If you try to delete a built-in toolbar, DELETE.TOOLBAR returns the #VALUE! error value, interrupts the macro, and takes no other action.

Related Functions

ADD.TOOL	Adds or more buttons to a toolbar
ADD.TOOLBAR	Creates a new toolbar with the specified buttons
RESET.TOOLBAR	Resets a built-in toolbar to its initial default setting

DEMOTE

Equivalent to clicking the Group tool. Demotes, or groups, the selected rows or columns in an outline. Use DEMOTE to change the configuration of an outline by grouping rows or columns of information.

Syntax

DEMOTE(**row_col**)

DEMOTE?(row_col)

Row_col specifies whether to group rows or columns.

Row_col	**Demotes**
1 or omitted	Rows
2	Columns

Remarks

- If the selection consists of an entire row or rows, then rows are demoted even if row_col is 2. Similarly, selection of an entire column overrides row_col 1.
- If the selection is unambiguous (an entire row or column), then DEMOTE? will not display the dialog box.

Related Functions

PROMOTE	Promotes the selection in an outline
SHOW.DETAIL	Expands or collapses a portion of an outline
SHOW.LEVELS	Displays a specific number of levels of an outline

DEREF

Returns the value of the cells in a reference.

Syntax

DEREF(**reference**)

Reference is the cell or cells from which you want to obtain a value. If reference is the reference of a single cell, DEREF returns the value of that cell. If reference is the reference of a range of cells, DEREF returns the array of values in those cells. If reference refers to the active sheet, it must be an absolute reference. Relative references are converted to absolute references.

Remarks

In most formulas, there is no difference between using a value and using the reference of a cell containing that value. The reference is automatically converted to the value, as necessary. For example, if cell A1 contains the value 2, then the formula =A1+1, like the formula =2+1, returns the result 3, because the reference A1 is converted to the value 2. However, in a few functions, such as the SET.NAME function, references are not automatically converted to values. Instead, those functions behave differently depending on whether an argument is a reference or a value.

Example

See the sixth example for SET.NAME.

Related Function

SET.NAME   Defines a names as a value

DESCR

Generates descriptive statistics for data in the input range.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

DESCR(**inprng**, outrng, grouped, labels, summary, ds_large, ds_small, confid)

DESCR?(inprng, outrng, grouped, labels, summary, ds_large, ds_small, confid)

Inprng   is the input range.

Outrng   is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Grouped   is a text character that indicates whether the data in the input range is organized by row or column.

- If grouped is "C" or omitted, then the data is organized by column.
- If grouped is "R" then the data is organized by row.

Labels   is a logical value that describes where the labels are located in the input range, as shown in the following table:

Labels	**Grouped**	**Labels are in**
TRUE	"C"	First row of the input range.
TRUE	"R"	First column of the input range.
FALSE or omitted	(ignored)	No labels. All cells in the input range are data.

Summary is a logical value. If TRUE, DESCR reports the summary statistics. If FALSE or omitted, no summary statistics are reported.

Ds_large is an integer k. If ds_large is present, DESCR reports the k-th largest data point. If ds_large is omitted, the value is not reported.

Ds_small is an integer k. If ds_small is present, DESCR reports the k-th smallest data point. If ds_small is omitted, the value is not reported.

Confid is the confidence level of the mean. If confid is given, DESCR reports the confidence interval for the input range. If confid is omitted, the confidence interval is 95%.

DIALOG.BOX

Displays the dialog box described in a dialog box definition table.

Syntax

DIALOG.BOX(**dialog_ref**)

Dialog_ref is a reference to a dialog box definition table on sheet, or an array containing the definition table.

- If an OK button in the dialog box is chosen, DIALOG.BOX enters values in fields as specified in the dialog_ref area and returns the position number of the button chosen. The position numbers start with 1 in the second row of the dialog box definition table.
- If the Cancel button in the dialog box is chosen, DIALOG.BOX returns FALSE.

The dialog box definition table must be at least seven columns wide and two rows high. The definitions of each column in a dialog box definition table are listed in the following table.

Column type	**Column number**
Item number	1
Horizontal position	2
Vertical position	3
Item width	4
Item height	5
Text	6

Initial value or result	7

The first row of dialog_ref defines the position, size, and name of the dialog box. It can also specify the default selected item and the reference for the Help button. The position is specified in columns 2 and 3, the size in columns 4 and 5, and the name in column 6. To specify a default item, place the item's position number in column 7. You can place the reference for the Help button in row 1, column 1 of the table, but the preferred location is column 7 in the row where the Help button is defined. Row 1, column 1 is usually left blank.

The following table lists the numbers for the items you can display in a dialog box.

Dialog-box item	**Item number**
Default OK button	1
Cancel button	2
OK button	3
Default Cancel button	4
Static text	5
Text edit box	6
Integer edit box	7
Number edit box	8
Formula edit box	9
Reference edit box	10
Option button group	11
Option button	12
Check box	13

Group box	14
List box	15
Linked list box	16
Icons	17
Linked file list box (Windows only)	18
Linked drive and directory box (Windows only)	19
Directory text box	20
Drop-down list box	21
Drop-down combination edit/list box	22
Picture button	23
Help button	24

Remarks

- Add 100 to an item number in the above table to define the item as a trigger. A trigger is a dialog box item that, when chosen, returns to your macro (as clicking OK would) but continues to display the dialog box, allowing your macro to change the dialog box definition or display an alert message or another dialog box. The Help button, edit boxes, group boxes, static text, and icons cannot be triggers.

- Add 200 to an item number to define it as dimmed. A dimmed (gray) item cannot be chosen or selected. For example, 203 is a dimmed OK button. You can use item 223 to include a picture in your dialog box that does not behave like a button.

- If a trigger has been chosen and you still want to clear a dynamic dialog box from the screen, use DIALOG.BOX(FALSE). This is useful if you want to confirm that the dialog box has been filled out correctly before dismissing it.

- The dialog box definition table can be an array. If dialog_ref is an array instead of a reference, DIALOG.BOX returns a modified copy of that array, along with the results of the dialog box in the seventh column. (The first item in the seventh column is the position number of the chosen button or of a triggered item.) This is useful if you want to preserve the original dialog box definition table since DIALOG.BOX does not modify the original array argument. If you cancel the dialog box, or if a dialog box error occurs, DIALOG.BOX returns FALSE instead of an array.

Related Functions

ALERT	Displays a dialog box and a message
INPUT	Displays a dialog box for user input

DIRECTORY

Sets the current drive and directory or folder to the specified path and returns the name of the new directory or folder as text. Use DIRECTORY to get the name of the current directory or folder for use with the OPEN and SAVE.AS functions or to specify a directory or folder from which to return a list of files with the FILES function.

Syntax

DIRECTORY(**path_text**)

Path_text is the drive and directory or folder you want to change to.

- If path_text is not specified, DIRECTORY returns the name of the current directory or folder as text.
- If path_text does not specify a drive, the current drive is assumed.

Examples

In Microsoft Excel for Windows, the following macro formula sets the directory to \EXCEL\MODELS on the current drive and returns the value "drive:\EXCEL\MODELS":

```
DIRECTORY("\EXCEL\MODELS")
```

The following macro formula sets the current drive to E and sets the directory to \EXCEL\MODELS on E. It returns the value "E:\EXCEL\MODELS":

```
DIRECTORY("E:\EXCEL\MODELS")
```

In Microsoft Excel for the Macintosh, the following macro formula sets the folder to HARD DISK: APPS:EXCEL:FINANCIALS and returns the value "HARD DISK:APPS:EXCEL:FINANCIALS":

```
DIRECTORY("HARD DISK:APPS:EXCEL:FINANCIALS")
```

Related Function

FILES	Returns the filenames in the specified directory or folder

DISABLE.INPUT

Blocks all input from the keyboard and mouse to Microsoft Excel (except input to displayed dialog boxes). Use DISABLE.INPUT to prevent input from the user or from other applications.

Syntax

DISABLE.INPUT(**logical**)

Logical is a logical value specifying whether input is currently disabled. If logical is TRUE, input is disabled; if FALSE, input is reenabled.

Remarks

Disabling input can be useful if you are using dynamic data exchange (DDE) to communicate with Microsoft Excel from another application.

Important Be sure to end any macro that uses DISABLE.INPUT(TRUE) with a DISABLE.INPUT(FALSE) function. If you do not include DISABLE.INPUT(FALSE) to allow non-dialog-box input, you will not be able to take any actions on your computer after the macro has finished.

Related Functions

CANCEL.KEY	Disables macro interruption
ENTER.DATA	Turns Data Entry mode on and off
WORKSPACE	Changes workspace settings

DISPLAY

Controls whether the screen displays formulas, gridlines, row and column headings, and other screen attributes. There are two syntax forms of this function. Use syntax 1 to control screen display. Use syntax 2 to control the display of the Info Window.

Syntax 1	Controls screen display
Syntax 2	Controls display of Info Window

DISPLAY SYNTAX 1

Controls whether the screen displays formulas, gridlines, row and column headings, and other screen attributes. There are two syntax forms of this function. Use syntax 1 to control screen display. This function is provided for compatibility with Microsoft Excel version 4.0. To control screen display in Microsoft Excel version 5.0 or later, see OPTIONS.VIEW.

Arguments for this syntax form correspond to options and check boxes in the Display Options dialog box in Microsoft Excel version 4.0. Arguments that correspond to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, no action is taken.

Syntax

DISPLAY(formulas, gridlines, headings, zeros, color_num, reserved, outline, page_breaks, object_num)

DISPLAY?(formulas, gridlines, headings, zeros, color_num, reserved, outline, page_breaks, object_num)

Formulas corresponds to the Formulas check box. The default is FALSE on worksheets and TRUE on macro sheets.

Gridlines corresponds to the Gridlines check box. The default is TRUE.

Headings corresponds to the Row & Column Headings check box. The default is TRUE.

Zeros corresponds to the Zero Values check box. The default is TRUE.

Color_num is a number from 0 to 56 corresponding to the gridline and heading colors in the Display Options dialog box; 0 corresponds to automatic color and is the default value.

Reserved is reserved for certain international versions of Microsoft Excel.

Outline corresponds to the Outline Symbols check box. The default is TRUE.

Page_breaks corresponds to the Automatic Page Breaks check box. The default is FALSE.

Object_num is a number from 1 to 3 corresponding to the display options in the Object box.

Object_num	**Corresponds to**
1 or omitted	Show All
2	Show Placeholders
3	Hide

Related Functions

OPTIONS.VIEW Controls display

WORKSPACE Changes workspace settings

ZOOM Enlarges or reduces a sheet in the active window

Syntax 2 Controls display of Info Window

DISPLAY SYNTAX 2

Equivalent to clicking the commands on the Info menu when the Info Window is active. Controls which commands on the Info Window are in effect. There are two syntax forms of this function. Use syntax 2 to control the display of the Info Window. The Info Window must be active to use this form of DISPLAY. This function is included for compatibility with Microsoft Excel 95 or earlier; the Info Window is not available in Microsoft Excel 97 or later.

Arguments in this syntax form correspond to commands on the Info menu with the same names.

For these arguments:

- If the argument is TRUE, Microsoft Excel displays the corresponding Info item.

- If the argument is FALSE, Microsoft Excel does not display the corresponding Info item.
- If the argument is omitted, the status of the item is unchanged.

Syntax

For controlling Info Window display

DISPLAY(cell, formula, value, format, protection, names, precedents, dependents, note)

Cell is a logical value that corresponds to the Cell command and controls the display of cell information in the Info Window. If TRUE, cell information will be displayed; if FALSE, cell information will not be displayed.

Formula is a logical value that corresponds to the Formula command and controls the display of formula information in the Info Window. If TRUE, formula information will be displayed; if FALSE, formula information will not be displayed.

Value is a logical value that corresponds to the Value command and controls the display of value information in the Info Window. If TRUE, value information will be displayed; if FALSE, value information will not be displayed.

Format is a logical value that corresponds to the Format command and controls the display of format information in the Info Window. If TRUE, format information will be displayed; if FALSE, format information will not be displayed.

Protection is a logical value that corresponds to the Protection command and controls the display of protection information in the Info Window. If TRUE, protection information will be displayed; if FALSE, protection information will not be displayed.

Names is a logical value that corresponds to the Names command and controls the display of name information in the Info Window. If TRUE, name information will be displayed; if FALSE, name information will not be displayed.

Precedents is a number from 1 to 3 that specifies which precedents to list, according to the following table.

Dependents is a number from 1 to 3 that specifies which dependents to list, according to the following table.

Precedents or dependents	List
0	None
1	Direct only
2	All levels

Note is a logical value that corresponds to the Note command and controls the display of note information in the Info Window. If TRUE, note information will be displayed; if FALSE, note information will not be displayed.

Related Functions

SHOW.INFO	Controls the display of the Info Window
ZOOM	Enlarges or reduces a sheet in the active window
Syntax 1	Controls screen display

DOCUMENTS

Returns, as a horizontal array in text form, the names of the specified open workbooks. Use DOCUMENTS to retrieve the names of open workbooks to use in other functions that manipulate open workbooks.

Syntax

DOCUMENTS(type_num, match_text)

Type_num is a number specifying whether to include add-in workbooks in the array of workbooks, according to the following table.

Type_num	**Returns**
1 or omitted	Names of all open workbooks except add-in workbooks
2	Names of add-in workbooks only
3	Names of all open workbooks

Match_text specifies the workbooks whose names you want returned and can include wildcard characters. If match_text is omitted, DOCUMENTS returns the names of all open workbooks.

Remarks

- Use the INDEX function to select individual workbook names from the array to use in other functions that take workbook names as arguments.
- Use COLUMNS to count the number of entries in the horizontal array.
- Use TRANSPOSE to change a horizontal array to a vertical one.
- Since the DOCUMENTS function only returns actual workbook names, it ignores any changes made by the WINDOW.TITLE function.

Examples

In Microsoft Excel for Windows, if your workspace contains windows named BUDGET.XLS, CHART1, ACTUAL.XLS:1, ACTUAL.XLS:2, and BOOK.XLS, then:

`DOCUMENTS (1)` equals the four-cell array {"ACTUAL.XLS", "BOOK.XLS", "BUDGET.XLS", "CHART1"}

SET.NAME("Document_array", DOCUMENTS()) defines the name, Document_array, as {"ACTUAL.XLS", "BOOK.XLS", "BUDGET.XLS", "CHART1"}

In Microsoft Excel for the Macintosh, if your workspace contains windows named BUDGET CHART1, ACTUALS, ACTUALS:2, and BOOK then:

DOCUMENTS(1) equals the four-cell array {"ACTUALS", "BOOK", "BUDGET", "CHART1"}

Related Functions

FILES   Returns the filenames in the specified directory or folder

GET.DOCUMENT   Returns information about a workbook

GET.WINDOW   Returns information about a window

WINDOWS   Returns the names of all open windows

DUPLICATE

Duplicates the selected object. If an object is not selected, returns the #VALUE! error value and interrupts the macro.

Syntax

DUPLICATE( )

Related Functions

COPY   Copies and pastes data or objects

PASTE   Pastes cut or copied data

ECHO

Controls screen updating while a macro is running. If a large macro uses many commands that update the screen, use ECHO to make the macro run faster.

Syntax

ECHO(logical)

Logical   is a logical value specifying whether screen updating is on or off.

- If logical is TRUE, Microsoft Excel selects screen updating.
- If logical is FALSE, Microsoft Excel clears screen updating.
- If logical is omitted, Microsoft Excel changes the current screen update condition.

Remarks

- Screen updating is always turned back on when a macro ends.
- You can use GET.WORKSPACE to determine whether screen updating is on or off.

Related Function

GET.WORKSPACE   Returns information about the workspace

EDITBOX.PROPERTIES

Sets the properties of an edit box on a dialog sheet.

Syntax

EDITBOX.PROPERTIES(validation_num, multiline_logical, vscroll_logical, password_logical)

EDITBOX.PROPERTIES?(validation_num, multiline_logical, vscroll_logical, password_logical)

Validation_num   is the validation applied to the edit box when the dialog is dismissed. If the edit box contains a value other than the type specified (or validation), an error is returned.

Validation_num	**Type**
1	Text
2	Integer
3	Number (allows floating point)
4	Reference
5	Formula

Multiline_logical   is a logical value specifying whether word wrapping is allowed in the edit box control. If TRUE, word wrapping is allowed. If FALSE, word wrapping is not allowed

Vscroll_logical   is a logical value specifying whether edit box displays a vertical scrollbar. If TRUE, a scrollbar is displayed. If FALSE, a scrollbar is not displayed.

Password_logical   is a logical value specifying whether edit box displays characters as the user types. If TRUE, asterisks (*) are displayed as the user types. If FALSE, no asterisks are displayed.

Related Functions

CHECKBOX.PROPERTIES   Sets various properties of check box and option box controls

PUSHBUTTON.PROPERTIES   Sets the properties of the push button control

EDIT.COLOR

Equivalent to clicking the Modify button on the Color tab, which appears when you click the Options command on the Tools menu. Defines the color for one of the 56 color palette boxes.

Use EDIT.COLOR if you want to use a color that is not currently on the palette and if your system hardware has more than 56 colors available. After you set the color for the color box, any items previously formatted with that color are displayed in the new color.

Syntax

EDIT.COLOR(**color_num**, red_value, green_value, blue_value)

EDIT.COLOR?(**color_num**, red_value, green_value, blue_value)

Color_num is a number from 1 to 56 specifying one of the 56 color palette boxes for which you want to set the color.

Red_value, green_value, and blue_value are numbers that specify how much red, green, and blue are in each color.

- In Microsoft Excel for Windows, red_value, green_value, and blue_value are numbers from 0 to 255.
- In Microsoft Excel for the Macintosh, red_value, green_value, and blue_value are also numbers from 0 to 255. However, the color editing dialog box displays numbers from 0 to 65, 535. Microsoft Excel automatically converts the numbers between the two ranges. This allows you to display similar colors in all operating environments without modifying your macros.
- If red_value, green_value, and blue_value are all set to 255, the resulting color is white. If they are all set to zero, the resulting color is black.
- If red_value, green_value, or blue_value is omitted, Microsoft Excel assumes it to be the appropriate value for that color_num.

Remarks

- Your system hardware determines the number of unique colors that you can choose from and the number of colors that can be displayed on the screen at the same time.
- EDIT.COLOR does not use hue, saturation, or brightness values. If you are using the macro recorder and set the color of a color palette box using hue, saturation, and luminance, Microsoft Excel records the corresponding red, green, and blue values instead.
- The dialog-box form of this function, EDIT.COLOR?(color_num), displays your system's color editing dialog box. The default red_value, green_value, and blue_value are determined by the current settings for the color_num you specify. Color_num is a required argument for the dialog-box form of this function.

Related Function

COLOR.PALETTE Copies a color palette from one workbook to another

EDIT.DELETE

Equivalent to clicking the Delete command on the Edit menu. Removes the selected cells from the worksheet and shifts other cells to close up the space.

Syntax

EDIT.DELETE(shift_num)

EDIT.DELETE?(shift_num)

Shift_num is a number from 1 to 4 specifying whether to shift cells left or up after deleting the current selection or else to delete the entire row or column.

Shift_num	Result
1	Shifts cells left
2	Shifts cells up
3	Deletes entire row
4	Deletes entire column

- If shift_num is omitted and if one cell or a horizontal range is selected, EDIT.DELETE shifts cells up.
- If shift_num is omitted and a vertical range is selected, EDIT.DELETE shifts cells left.

Related Function

CLEAR Clears specified information from the selected cells or chart

EDITION.OPTIONS

Sets options in, or performs actions on, the specified publisher or subscriber. In Microsoft Excel for Windows, EDITION.OPTIONS also allows you to cancel a publisher or subscriber created in Microsoft Excel for the Macintosh.

Syntax

EDITION.OPTIONS(**edition_type**, edition_name, reference, **option**, appearance, size, formats)

Edition_type is the number 1 or 2 specifying the type of edition.

Edition_type	Type of edition

1	Publisher
2	Subscriber

Edition_name is the name of the edition you want to change the edition options for or to perform actions on. If edition_name is omitted, reference is required.

Reference specifies the range (given in text form as a name or an R1C1-style reference) occupied by the publisher or subscriber.

- Reference is required if you have more than one publisher or subscriber of edition_name on the active workbook. Use reference to specify the location of the publisher or subscriber for which you want to set options.
- If edition_type is 1 and the publisher is an embedded chart, or if edition_type is 2 and the subscriber is a picture, reference is the object identifier as displayed in the reference area.
- If reference is omitted, edition_name is required.

Option is a number from 1 to 6 specifying the edition option you want to set or the action you want to take, according to the following two tables. Options 2 to 6 are only available if you are using Microsoft Excel for the Macintosh with system software version 7.0 or later.

If a publisher is specified, then option applies as follows.

Option	**Action**
1	Cancels the publisher
2	Sends the edition now
3	Selects the range or object published to the specified edition
4	Automatically updates the edition when the file is saved
5	Updates the edition on request only
6	Changes the edition file as specified by appearance, size, and formats

If a subscriber is specified, then option applies as follows.

Option	Action
1	Cancels the subscriber
2	Gets the latest edition
3	Opens the publisher workbook
4	Automatically updates when new data is available
5	Update on request only

The following three arguments are available only when option is 6.

Appearance specifies whether the selection is published as shown on screen or as shown when printed. The default value for appearance is 1 if the selection is a sheet or macro sheet and 2 if the selection is a chart.

Appearance	Selection is published
1	As shown on screen
2	As shown when printed

Size specifies the size of a published chart. Size is only available if a chart is to be published.

Size	Chart size is published
1 or omitted	As shown on screen
2	As shown when printed

Formats is a number specifying the format of the file.

Formats	File format

1 or omitted	PICT
2	BIFF
4	RTF
8	VALU

You can also use the sum of the allowable file formats. For example, a value of 6 specifies BIFF and RTF.

Example

The following macro formula opens the workbook (and application) that published the edition named Monthly Totals:

```
EDITION.OPTIONS(2, "Monthly Totals", , 3)
```

Related Functions

CREATE.PUBLISHER	Creates a publisher from the selection
GET.LINK.INFO	Returns information about a link
SUBSCRIBE.TO	Inserts contents of an edition into the active workbook

EDIT.OBJECT

Equivalent to clicking the Edit command on the (selected object) Object submenu of the Edit menu. Starts the application associated with the selected object and makes the object available for editing or other actions.

Syntax

EDIT.OBJECT(verb_num)

Verb_num is a number specifying which verb to use while working with the object, that is, what you want to do with the object.

- The available verbs are determined by the object's source application. 1 often specifies "edit, " and 2 often specifies "play" (for sound, animation, and so on). For more information, consult the documentation for the object's application to see how it supports object linking and embedding (OLE).
- If the object does not support multiple verbs, verb_num is ignored.
- If verb_num is omitted, it is assumed to be 1.

Remarks

Your macro pauses while you're editing the object and resumes when you return to Microsoft Excel.

Related Function

INSERT.OBJECT   Creates an object of a specified type

EDIT.REPEAT

Equivalent to clicking the Repeat command on the Edit menu. Repeats certain actions and commands. EDIT.REPEAT is available in the same situations as the Repeat command.

Syntax

EDIT.REPEAT( )

EDIT.SERIES

Equivalent to clicking the Edit Series command on the Chart menu in Microsoft Excel version 4.0. Creates or changes chart series by adding a new SERIES formula or modifying an existing SERIES formula in the topmost chart type. Chart types are displayed in the following order from top to bottom: XY (Scatter), Line, Column, Bar, Area.

Syntax

EDIT.SERIES(**series_num**, **name_ref**, **x_ref**, **y_ref**, **z_ref**, **plot_order**)

EDIT.SERIES?(series_num, name_ref, x_ref, y_ref, z_ref, plot_order)

Series_num   is the number of the series you want to change. If series_num is 0 or omitted, Microsoft Excel creates a new data series.

Name_ref   is the name of the data series. It can be an external reference to a single cell, a name defined as a single cell, or a name defined as a sequence of characters. Name_ref can also be text (for example, "Projected Sales").

X_ref   is an external reference to the name of the sheet and the cells that contain one of the following sets of data:

- Category labels for all charts except xy (scatter) charts
- X-coordinate data for xy (scatter) charts

Y_ref   is an external reference to the name of the sheet and the cells that contain values (or y-coordinate data in xy (scatter) charts) for all 2-D charts. Y_ref is required in 2-D charts but does not apply to 3-D charts.

Z_ref   is an external reference to the name of the sheet and the cells that contain values for all 3-D charts. Z_ref is required in 3-D charts but does not apply to 2-D charts.

Plot_order   is a number specifying whether the data series is plotted first, second, and so on, in the chart type.

- If you assign a plot_order to a series, Microsoft Excel plots that series in the order you specify, and the series that previously had that plot order (and any series following it) has its plot order increased by one.

- If you add a series to a chart with an overlay, the number of series in the main chart does not change, so if the series is added to the main chart, then the series that was plotted last in the main chart will be plotted first in the overlay chart. To change which series is plotted first in the overlay chart, use the (chart type) Group command from the Format menu, and then select the Series Order tab in the Format (chart type) Group dialog box. You can also use the FORMAT.OVERLAY function.

- If you omit plot_order when you add a new series, then Microsoft Excel plots that series last and assigns it the correct plot_order value.

- The maximum value for plot_order is 255.

Remarks

To change where a series is plotted within a chart, you can change the chart type, using the FORMAT.CHART function, or the plot order. Plot order affects where the series appears within the chart type only.

X_ref, y_ref, and z_ref can be arrays or references to a nonadjacent selection, although they cannot be names that refer to a nonadjacent selection. If you specify a nonadjacent selection for any of these arguments, make sure to enclose the reference to the selection in parentheses so that Microsoft Excel does not treat the components of the references as separate arguments.

Tip To delete a data series, use the SELECT("Sn") macro function, where n is the series number, followed by the FORMULA("") macro function. You can also use the CLEAR function instead of FORMULA.

Related Function

FORMAT.CHART

EDIT.TOOL

Displays the Button Editor dialog box, which you use to change the appearance of a button on a toolbar.

Syntax

EDIT.TOOL(**bar_id**, **position**)

Bar_id is the number of the toolbar containing the button you want to edit. For a list of toolbar numbers, see ADD.TOOL. Use the GET.TOOLBAR function to return the information about a toolbar.

Position is the position on the toolbar of the button you want to edit. Buttons are numbered from the left starting at 1. Gaps between buttons are counted as positions.

Related Functions

ADD.TOOL Adds a button to a toolbar

GET.TOOLBAR Returns information about a toolbar

ELSE

Used with IF, ELSE.IF, and END.IF to control which functions are carried out in a macro. ELSE signals the beginning of a group of formulas in a macro sheet that will be carried out if the results of all preceding ELSE.IF statements and the preceding IF statement are FALSE. Use ELSE with IF, ELSE.IF, and END.IF when you want to perform multiple actions based on a condition. This method is preferable to using GOTO because it makes your macros more structured.

Syntax

ELSE( )

Remarks

ELSE must be entered in a cell by itself. In other words, the cell can contain only "=ELSE()".

For more information about ELSE, ELSE.IF, END.IF, and IF, and for examples of these functions, see form 2 of the IF function.

Related Functions

ELSE.IF   Specifies an action to take if an IF or another ELSE.IF function returns FALSE

END.IF   Ends a group of macro functions started with an IF statement

IF   Specifies an action to take if a logical test is TRUE

ELSE.IF

Used with IF, ELSE, and END.IF to control which functions are carried out in a macro. ELSE.IF signals the beginning of a group of formulas in a macro sheet that will be carried out if the preceding IF or ELSE.IF function returns FALSE and if logical_test is TRUE. Use ELSE.IF with IF, ELSE, and END.IF when you want to perform multiple actions based on a condition. This method is preferable to using GOTO because it makes your macros more structured.

Syntax

ELSE.IF(**logical_test**)

Logical_test   is a logical value that ELSE.IF uses to determine what functions to carry out next—that is, where to branch.

- If logical_test is TRUE, Microsoft Excel carries out the functions between the ELSE.IF function and the next ELSE.IF, ELSE, or END.IF function.
- If logical_test is FALSE, Microsoft Excel immediately branches to the next ELSE.IF, ELSE, or END.IF function.

Remarks

- ELSE.IF must be entered in a cell by itself.
- Logical_test will always be evaluated, even if the ELSE.IF section is not reached (due to a previous IF or ELSE.IF logical_test evaluating to TRUE). For this reason, you should not use formulas that carry out actions for logical_test. If you need to base the ELSE.IF condition on the return value of a formula that carries out an

action, use the form "ELSE, IF(logical_test), and END.IF" in place of "ELSE.IF(logical_test)."

For more information about ELSE, ELSE.IF, END.IF, and IF, and for examples of these functions, see form 2 of the IF function.

Related Functions

ELSE   Specifies an action to take if an IF function returns FALSE

END.IF   Ends a group of macro functions started with an IF statement

IF   Specifies an action to take if a logical test is TRUE

EMBED

Displayed in the formula bar when an embedded object is selected. EMBED cannot be entered on a sheet or used in a macro.

Syntax

EMBED(**object_class**, item)

Object_class   is the name of the application and document type that created the embedded object. For example, the object_class arguments used when Microsoft Excel sheets are embedded in other applications are "Excel.sheet.5" and "Excel.Chart.5".

Item   is the area selected to copy, and determines the view on the embedded document. When item is empty text (""), EMBED creates a view on the entire document.

Remarks

If you delete the EMBED formula, the embedded object remains on the sheet as a graphic, and the link to the creating application is deleted. Double-clicking the object no longer starts the creating application.

ENABLE.COMMAND

Enables or disables a custom command or menu. Disabled commands appear dimmed and can't be chosen. Use ENABLE.COMMAND to control which commands the user can click in a menu bar.

Syntax

ENABLE.COMMAND(**bar_num**, **menu**, **command**, **enable**, subcommand)

Bar_num   is the menu bar in which a command resides. Bar_num can be the number of a built-in menu bar or the number returned by a previously run ADD.BAR function. See ADD.COMMAND for a list of the built-in menu bar numbers.

Menu   is the menu on which the command resides. Menu can be either the name of a menu as text or the number of a menu. Menus are numbered starting with 1 from the left of the screen.

Command   is the command you want to enable or disable. Command can be either the name of the command as text or the number of the command. The top command on a menu is command 1. If command is 0, ENABLE.COMMAND enables or disables the entire menu.

Enable is a logical value specifying whether the command should be enabled or disabled. If enable is TRUE, Microsoft Excel enables the command; if FALSE, it disables the command.

Subcommand is the name of the command on a submenu that you want to enable. If you use subcommand, you must use command as the name of the submenu. Use subcommand 0 to enable an entire submenu.

Remarks

- You cannot disable built-in commands. If the specified command is a built-in command or does not exist, ENABLE.COMMAND returns the #VALUE! error value and interrupts the macro.
- You can hide any shortcut menu from users by using ENABLE.COMMAND with command set to 0.

Example

The following macro formula disables a custom command that had been added previously to the View menu on the worksheet and macro sheet menu bar:

```
ENABLE.COMMAND(10, "View", "Audit...", FALSE)
```

Related Functions

ADD.BAR	Adds a menu bar
ADD.COMMAND	Adds a command to a menu
CHECK.COMMAND	Adds or deletes a check mark to or from a command
DELETE.COMMAND	Deletes a command from a menu
RENAME.COMMAND	Changes the name of a command or menu

ENABLE.OBJECT

Enables or disables a drawing object or the selected drawing object. A disabled object will not run any macro events assigned to it, and the controls will be grayed out.

Syntax

ENABLE.OBJECT(object_id_text, **enable_logical**)

Object_id_text is the name of the object(s) as text. If omitted, the selected object(s) are assumed.

Enable_logical is a logical value that specifies whether the object is to be enabled. If TRUE, the object is enabled. If FALSE, the object is disabled.

Examples

`ENABLE.OBJECT("Button 2",FALSE)` disables the button with object name Button 2 on the dialog box.

Related Function

SET.CONTROL.VALUE	Changes the value of the active control

ENABLE.TIPWIZARD

This function should not be used. The TipWizard has been removed from Microsoft Excel.

ENABLE.TOOL

Enables or disables a button on a toolbar. An enabled button can be accessed by the user. Disabled buttons may still be visible but cannot be accessed. Use ENABLE.TOOL to control which buttons the user can click in a particular situation.

Syntax

ENABLE.TOOL(**bar_id, position**, enable)

Bar_id is the number or name of a toolbar on which the button resides. For detailed information about bar_id, see ADD.TOOL.

Position specifies the position of the button on the toolbar. Position starts with 1 at the left side (if horizontal) or from the top (if vertical).

Enable specifies whether the button can be accessed. If enable is TRUE or omitted, the user can access the button; if FALSE, the user cannot access it.

Remarks

Microsoft Excel sounds a tone if you click a disabled button.

Example

The following macro formula enables the fourth button in Toolbar1:

```
ENABLE.TOOL("Toolbar1", 4, TRUE)
```

Related Function

GET.TOOL	Returns information about a button or buttons on a toolbar

END.IF

Ends a block of functions associated with the preceding IF function. You must include one and only one END.IF function for each macro-sheets-only syntax form (syntax 2) of the IF function in a macro. Syntax 1 of the IF function, which can be used on both worksheets and macro sheets, does not require an END.IF function. Use END.IF with IF, ELSE, and ELSE.IF when you want to perform multiple actions based on a condition. This method is preferable to using GOTO because it makes your macros more structured.

Syntax

END.IF( )

Remarks

- If you accidentally omit an END.IF function, your macro will end with an error at the cell containing the first IF function that does not have a corresponding END.IF function.
- END.IF must be entered in a cell by itself.

- For more information about ELSE, ELSE.IF, END.IF, and IF, and for examples of these functions, see form 2 of the IF function.

Related Functions

ELSE Specifies an action to take if an IF function returns FALSE

ELSE.IF Specifies an action to take if an IF or another ELSE.IF function returns FALSE

IF Specifies an action to take if a logical test is TRUE

ENTER.DATA

Turns on Data Entry mode and allows you to select and to enter data into the unlocked cells in the current selection only (the data entry area). Use ENTER.DATA when you want to enter data only in a specific part of your sheet. You can then use that part of the sheet as a simple data form.

Syntax

ENTER.DATA(logical)

Logical is a logical value that turns Data Entry mode on or off.

- If logical is TRUE, Data Entry mode is turned on; if FALSE, Data Entry mode is turned off and data entry, cell movement, and cell selection return to normal. If logical is omitted, ENTER.DATA toggles Data Entry mode.
- Logical can also be the number 2. This setting turns on Data Entry mode and prevents the ESC key from turning it off.
- Logical can also be a reference. Using a reference for this argument turns on Data Entry mode for the supplied reference.

Remarks

- In Data Entry mode, you can move the active cell and select cell ranges only in the data entry area. The arrow keys and the TAB and SHIFT+TAB keys move from one unlocked cell to the next. The HOME and END keys move to the first and last cell in the data entry area, respectively. You cannot select entire rows or columns, and clicking a cell outside the data entry area does not select it.
- The only commands available in Data Entry mode are commands normally available to protected workbooks.
- To turn off Data Entry mode, press ESC (unless logical is 2), activate another sheet in the active workbook window, or use another ENTER.DATA function. If you use another ENTER.DATA function, you will usually design your macros in one of two ways:
  - The macro turns on Data Entry mode, pauses while you enter data, resumes, and then turns off Data Entry mode.

- The macro turns on Data Entry mode and ends. After entering data, another macro turns off Data Entry mode; this latter macro could be assigned to a "Finished" button, for example.

With either method, you can use Microsoft Excel's ON functions to resume or run other macros based on an event, such as pressing the CONTROL+D keys.

Tips

- Normally you use Data Entry mode to enter data, but you can also prevent someone from entering data or moving the active cell by locking all the cells in the current selection before turning on Data Entry mode. This is useful if you want a user to view a range of cells but not change it or move the active cell. Similarly, if you unlock certain cells, you can restrict the user's movement to the Data Entry area only.

- To prevent someone from activating another workbook, which would turn off Data Entry mode, use the ON.WINDOW function or an Auto_Deactivate macro.

Related Functions

DISABLE.INPUT	Blocks all input to Microsoft Excel
FORMULA	Enters values into a cell or range or onto a chart

ERROR

Specifies what action to take if an error is encountered while a macro is running. Use ERROR to control whether Microsoft Excel error messages are displayed, or to run your own macro when an error is encountered.

Syntax

ERROR(**enable_logical**, macro_ref)

Enable_logical   is a logical value or number that selects or clears error-checking.

- If enable_logical is FALSE or 0, all error-checking is cleared. If error-checking is cleared and an error is encountered while a macro is running, Microsoft Excel ignores it and continues. Error-checking is selected again by an ERROR(TRUE) statement, or when the macro stops running.

- If enable_logical is TRUE or 1, you can either select normal error-checking (by omitting the other argument) or specify a macro to run when an error is encountered by using the macro_ref argument. When normal error-checking is active, the Macro Error dialog box is displayed when an error is encountered. You can halt the macro, start single-stepping through the macro, continue running the macro normally, or go to the macro cell where the error occurred.

- If enable_logical is 2 and macro_ref is omitted, error-checking is normal except that if the user clicks the Cancel button in an alert message, ERROR returns FALSE and the macro is not interrupted.

- If enable_logical is 2 and macro_ref is given, the macro goes to that macro_ref when an error is encountered. If the user clicks the Cancel button in an alert message, FALSE is returned and the macro is not interrupted.

Macro_ref specifies a macro to run if enable_logical is TRUE, 1, or 2 and an error is encountered. It can be either the name of the macro or a cell reference. If enable_logical is FALSE or 0, macro_ref is ignored.

Important Both ERROR(FALSE) and ERROR(TRUE, macro_ref) keep Microsoft Excel from displaying any messages at all, including the message asking whether to save changes when you close an unsaved workbook. If you want alert messages but not error messages to be displayed, use ERROR(2, macro_ref).

Remarks

You can use GET.WORKSPACE to determine whether error-checking is on or off.

Examples

`ERROR(FALSE)` clears error-checking.

`ERROR(TRUE, Recover)` selects error-checking and runs the macro named Recover when an error is encountered.

The following macro runs the macro ForceMenus if an error occurs in the current macro:

```
=ERROR(TRUE, ForceMenus)
```

Related Functions

CANCEL.KEY Disables macro interruption

LAST.ERROR Returns the reference of the cell where the last error occurred

ON.KEY Runs a macro when a specified key is pressed

ERRORBAR.X, ERRORBAR.Y

Adds error bars to the selected series in a chart. ERRORBAR.X adds bars showing the error factor for the X (category) axis and works for XY (scatter) charts only. ERRORBAR.Y adds bars showing the error factor for the Y (value) axis for all charts.

Syntax

ERRORBAR.X(include, type, amount, minus)

ERRORBAR.Y(include, type, amount, minus)

Include specifies the type of error value to include:

Include	Type of error value
1 or omitted	Plus and minus
2	Plus
3	Minus

4	None

Type specifies the type of error bars to display:

Type	Type of error displayed
1 or omitted	Fixed amount
2	Percent
3	Multiplying factor standard deviation (default value is 1)
4	Standard error
5	Custom

Amount is the range of error values to display. This argument depends on the value of type:

If type is	then amount
1 or omitted	Can be any number greater than 0
2	Can be any number greater than 0
3	Can be any number greater than or 0
4	Not required
5	Is the positive amount for custom error bars

Minus is the negative amount for custom error bars. Applicable only if type is 5.

Remarks

For the amount argument, standard deviation(s) can be calculated using this equation:

$$S.D.=\sqrt{\frac{\sum_{s=1}^{m}\sum_{i=1}^{n}{y_{is}}^{2}}{(n_{y}-1)}}$$

$$M=\frac{\sum_{s=1}^{m}\sum_{i=1}^{n}y_{is}}{n_{y}}$$

The standard deviation is multiplied by the value specified by amount and the error bars are placed this distance from the arithmetic mean. Therefore, these error bars are plotted along the arithmetic mean, not attached to data series.

Microsoft Excel calculates the standard error using the following equation:

$$S.E.=\sqrt{\frac{\sum_{s=1}^{m}\sum_{i=1}^{n}{y_{is}}^{2}}{(n_{y}-1)(n_{y})}}$$

Both the standard deviation and standard error functions use the following variables:

Variable	Equals
s	Series number
i	Point number in series s
m	Number of series for point y in chart
n	Number of points in each series
Yi	Data value of series s and the ith point
Ny	Total number of data values in all series
M	Arithmetic mean

EVALUATE

Evaluates a formula or expression that is in the form of text and returns the result. To run a macro or subroutine, use the RUN function.

Syntax

EVALUATE(**formula_text**)

Formula_text   is the expression in the form of text that you want to evaluate.

Remarks

Using EVALUATE is similar to selecting an expression within a formula in the formula bar and pressing the Recalculate key (F9 in Microsoft Excel for Windows and COMMAND+= in Microsoft Excel for the Macintosh). EVALUATE replaces an expression with a value.

Example

Suppose you want to know the value of a cell named LabResult1, LabResult2, or LabResult3, where the 1, 2, or 3 is specified by the name TrialNum whose value may change as the macro runs. You can use the following formula to calculate the value:

```
EVALUATE("LabResult"&TrialNum)
```

Related Function

RUN   Runs a macro

EXEC

Starts a separate program. Use EXEC to start other programs with which you want to communicate. Use EXEC with Microsoft Excel's other DDE functions (INITIATE, EXECUTE, and SEND.KEYS) to create a channel to another program and to send keystrokes and commands to the program. (SEND.KEYS is available only in Microsoft Excel for Windows.)

Syntax 1 is for Microsoft Excel for Windows. Syntax 2 is for Microsoft Excel for the Macintosh.

Syntax 1

For Microsoft Excel for Windows

EXEC(program_text, window_num)

Syntax 2

For Microsoft Excel for the Macintosh

EXEC(program_text, , background, preferred_size_only)

Important   Microsoft Excel for the Macintosh requires system software version 7.0 or later for the last two arguments of this function.

Program_text   is the name, as a text string, of any executable file or, in Microsoft Excel for Windows, any data file that is associated with an executable file.

- Use paths when the file or program to be started is not in the current directory or folder.

- In Microsoft Excel for Windows, program_text can include any arguments and switches that are accepted by the program to be started. Also, if program_text is the name of a file associated with a specific installed program, EXEC starts the program and loads the specified file.

Note In Microsoft Excel for the Macintosh, you must use an extra comma after the program_text argument. This skips the window_num argument that does not apply to the Macintosh.

Window_num is a number from 1 to 3 that specifies how the window containing the program should appear. Window_num is only available for use with Microsoft Excel for Windows. The window_num argument is allowed on the Macintosh, but it is ignored.

Window_num	**Window appears**
1	Normal size
2 or omitted	Minimized size
3	Maximized size

Background is a logical value that determines whether the program specified by program_text is opened as the active program or in the background, leaving Microsoft Excel as the active program. If background is TRUE, the program is started in the background; if FALSE or omitted, the program is started in the foreground. Background is only available for use with Microsoft Excel for the Macintosh and system software version 7.0 or later.

Preferred_size_only is a logical value that determines the amount of memory allocated to the program. If preferred_size_only is TRUE, the program is opened with its preferred memory allocation; if FALSE or omitted, it opens with the available memory if greater than its minimum requirement. Preferred_size_only is only available for use with Microsoft Excel for the Macintosh and system software version 7.0 or later. For information about changing the preferred memory size, see your Macintosh documentation.

Remarks

In Microsoft Excel for Windows and in Microsoft Excel for the Macintosh with system software version 7.0, if the EXEC function is successful, it returns the task ID number of the started program. The task ID number is a unique number that identifies a program. Use the task ID number in other macro functions, such as APP.ACTIVATE, to refer to the program. In Microsoft Excel for the Macintosh with system software version 6.0, if EXEC is successful, it returns TRUE. If EXEC is unsuccessful, it returns the #VALUE! error value.

Examples

In Microsoft Excel for Windows, the following macro formula starts the program SEARCH.EXE. Use paths when the file or program to be started is not in the current directory:

```
EXEC("C:\WINDOWS\SEARCH.EXE")
```

The following macro formula starts Microsoft Word for Windows and loads the document SALES.DOC:

```
EXEC("C:\WINWORD\WINWORD.EXE C:\MYFILES\SALES.DOC")
```

In Microsoft Excel for the Macintosh, the following macro formula starts Microsoft Word:

```
EXEC("HARD DISK:APPS:WORD")
```

Related Functions

APP.ACTIVATE   Switches to another application

EXECUTE   Carries out a command in another application

INITIATE   Opens a channel to another application

SEND.KEYS   Sends a key sequence to an application

TERMINATE   Closes a channel to another application

REQUEST   Requests an array of a specific type of information from an application with which you have a dynamic data exchange (DDE) link

POKE   Sends data to another application with which you have a dynamic data exchange (DDE) link

EXECUTE

Carries out commands in another program with which you have a dynamic data exchange (DDE) link. Use with EXEC, INITIATE, and SEND.KEYS to run another program through Microsoft Excel. (SEND.KEYS is available only in Microsoft Excel for Windows.)

Important   Microsoft Excel for the Macintosh requires system software version 7.0 or later for this function.

Syntax

EXECUTE(**channel_num, execute_text**)

Channel_num   is a number returned by a previously run INITIATE function. Channel_num refers to a channel through which Microsoft Excel communicates with another program.

Execute_text   is a text string representing commands you want to carry out in the program specified by channel_num. The form of execute_text depends on the program you are referring to. To include specific key sequences in execute_text, use the format described under key_text in the ON.KEY function.

If EXECUTE is not successful, it returns one of the following error values:

Value returned	**Situation**
#VALUE!	Channel_num is not a valid channel number.
#N/A	The program you are accessing is busy.

#DIV/0!	The program you are accessing does not respond after a certain length of time or you have pressed ESC to cancel.
#REF!	The keys specified in execute_text are refused by the application which you want to access.

Remarks

Commands sent to another program with EXECUTE will not work when a dialog box is displayed in the program. In Microsoft Excel for Windows, you can use SEND.KEYS to send commands that make selections in a dialog box.

Examples

The following macro formula sends the number 25 and a carriage return to the application identified by channel_num 14:

```
EXECUTE(14, "25~")
```

Related Functions

EXEC	Starts another application
INITIATE	Opens a channel to another application
POKE	Sends data to another application
REQUEST	Returns data from another application
SEND.KEYS	Sends a key sequence to an application
TERMINATE	Closes a channel to another application

EXPON

Predicts a value based on the forecast for the prior period, adjusted for the error in that prior forecast.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

EXPON(**inprng**, outrng, damp, stderrs, chart)

Inprng is the input range.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Damp is the damping factor. If omitted, damp is 0.3.

Stderrs is a logical value. If TRUE, standard error values are included in the output table. If FALSE, standard errors are not included.

Chart is a logical value. If TRUE, EXPON generates a chart for the actual and forecast values. If FALSE, the chart is not generated.

Related Function

MOVEAVG   Returns values along a moving average trend

EXTEND.POLYGON

Adds vertices to a polygon. This function must immediately follow a CREATE.OBJECT function or another EXTEND.POLYGON function. Use multiple EXTEND.POLYGON functions to create arbitrarily complex polygons. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax

EXTEND.POLYGON(**array**)

Array   is an array of values, or a reference to a range of cells containing values, that indicate the position of vertices in the polygon. The position is measured in points and is relative to the upper-left corner of the polygon's bounding rectangle.

- A vertex is a point. Each vertex is defined by a pair of coordinates in one row of array.
- The polygon is defined by the array argument to the CREATE.OBJECT function and to all the immediately following EXTEND.POLYGON functions.
- If the polygon contains many vertices, one array may not be sufficient to define it. If the number of elements in the formula exceeds 1024, you must include additional EXTEND.POLYGON functions. If you're recording a macro, Microsoft Excel automatically records additional EXTEND.POLYGON functions as needed.

Related Functions

CREATE.OBJECT   Creates an object

FORMAT.SHAPE   Inserts, moves, or deletes vertices of the selected polygon

EXTRACT

Equivalent to choosing the Extract command from the Data menu in Microsoft Excel version 4.0. Finds database records that match the criteria defined in the criteria range and copies them into a separate extract range.

Syntax

EXTRACT(unique)

EXTRACT?(unique)

Unique   is a logical value corresponding to the Unique Records Only check box in the Extract dialog box.

- If unique is TRUE, Microsoft Excel selects the check box and excludes duplicate records from the extract list.
- If unique is FALSE or omitted, Microsoft Excel clears the check box and extracts all records matching the criteria.

Related Functions

DATA.FIND   Finds records in a database

SET.CRITERIA   Defines the name Criteria for the selected range on the active sheet

SET.DATABASE   Defines the name Database for the selected range on the active sheet

SET.EXTRACT   Defines the name Extract for the selected range on the active sheet

FCLOSE

Closes the specified file.

Syntax

FCLOSE(**file_num**)

File_num   is the number of the file you want to close. File_num is returned by the FOPEN function that originally opened the file. If file_num is not a valid file number, FCLOSE halts the macro and returns the #VALUE! error value.

Examples

The following function closes the file identified by FileNumber:

```
FCLOSE(FileNumber)
```

Related Functions

CLOSE   Closes the active window

FILE.CLOSE   Closes the active workbook

FOPEN   Opens a file with the type of permission specified

FILE.CLOSE

Equivalent to clicking the Close command on the File menu. Closes the active workbook.

Syntax

FILE.CLOSE(save_logical, route_logical)

Save_logical   is a logical value specifying whether to save the file before closing it.

Save_logical	**Result**
TRUE	Saves the workbook
FALSE	Does not save the workbook

Omitted	If you've made changes to the workbook, displays a dialog box asking if you want to save the workbook

Route_logical is a logical value that specifies whether to route the file after closing it. This argument is ignored if there is not a routing slip present.

Route_logical	Result
TRUE	Routes the file
FALSE	Does not route the file

Omitted If you've specified recipients for routing, displays a dialog box asking if you want to save the file

Remarks

If you make any changes to the structure of a workbook, such as the name of sheets, their order, and so on, then a message will be displayed reminding you that there are unsaved changes, regardless of the save_logical value.

Note When you use the FILE.CLOSE function, Microsoft Excel does not run any Auto_Close macros before closing the workbook.

Related Functions

CLOSE Closes the active window

CLOSE.ALL Closes all unprotected windows

FCLOSE Closes a text file

FILE.DELETE

Deletes a file from the disk. Although you will normally delete files manually, you can, for example, use FILE.DELETE in a macro to delete temporary files created by the macro.

Syntax

FILE.DELETE(**file_text**)

FILE.DELETE?(file_text)

File_text is the name of the file to delete.

Remarks

- If Microsoft Excel can't find file_text, it displays a message saying that it cannot delete the file. To avoid this, include the entire path in file_text. See the following second and fifth examples. You can also use FILES to generate an array of filenames and then check if the file you want to delete is in the array.

- If a file is open when you delete it, the file is removed from the disk but remains open in Microsoft Excel.
- In the dialog-box form, FILE.DELETE?, you can use an asterisk (*) to represent any series of characters and a question mark (?) to represent any single character. See the following third and sixth examples.

Examples

In Microsoft Excel for Windows, the following macro formula deletes a file called CHART1.XLS from the current directory:

```
FILE.DELETE("CHART1.XLS")
```

The following macro formula deletes a file called 92INFO.XLS kept in the EXCEL\SALES subdirectory:

```
FILE.DELETE("C:\EXCEL\SALES\92INFO.XLS")
```

The following macro formula displays the Delete dialog box listing all documents whose extensions begin with the letters "XL":

```
FILE.DELETE?("*.XL?")
```

In Microsoft Excel for the Macintosh, the following macro formula deletes a file called CHART1 from the current folder:

```
FILE.DELETE("CHART1")
```

The following macro formula deletes a file called 1992 INFO kept in a series of nested folders:

```
FILE.DELETE("HARD DISK:EXCEL 5:SALES WORKSHEETS:1992 INFO")
```

The following macro formula displays the Delete dialog box listing all documents beginning with the word "Clients":

```
FILE.DELETE?("Clients*")
```

Related Functions

FILE.CLOSE   Closes the active workbook

FILES   Returns the filenames in the specified directory or folder

FILES

Returns a horizontal text array of the names of all files in the specified directory or folder. Use FILES to build a list of filenames upon which you want your macro to operate.

Syntax

FILES(directory_text)

Directory_text   specifies which directories or folders to return filenames from.

- Directory_text accepts an asterisk (*) to represent a series of characters and a question mark (?) to represent a single character in filenames.
- If directory_text is not specified, FILES returns filenames from the current directory.

Remarks

If you enter FILES in a single cell, only one filename is returned. You will normally use FILES with SET.NAME to assign the returned array to a name. See the last example below.

Tips You can use COLUMNS to count the number of entries in the returned array. You can use TRANSPOSE to change a horizontal array to a vertical one.

Examples

In Microsoft Excel for Windows, the following macro formula returns the names of all files starting with the letter F in the current directory or folder:

```
FILES("F*.*")
```

When entered as an array formula in several cells, the following macro formula returns the filenames in the current directory to those cells. If the directory contains fewer files than can fit in the selected cells, the #N/A error value appears in the extra cells.

```
FILES()
```

In Microsoft Excel for Windows, the following macro formula returns all files starting with "SALE" and ending with the .XLS extension in the \EXCEL\CHARTS subdirectory:

```
FILES("C:\EXCEL\CHARTS\SALE*.XLS")
```

In Microsoft Excel for the Macintosh, the following macro formula returns all files starting with "SALE" in the nested CHART folder:

```
FILES("DISK:EXCEL:CHART:SALE*")
```

The following macro stores the names of the files in the current directory in the named array FileArray

```
SET.NAME("FileArray",FILES())
```

Related Functions

DOCUMENTS   Returns the names of the specified open workbooks

FILE.DELETE   Deletes a file

OPEN   Opens a workbook

SET.NAME   Defines a name as a value

FILL.AUTO

Equivalent to copying cells or automatically filling a selection by dragging the fill selection handle with the mouse (the AutoFill feature).

Syntax

FILL.AUTO(destination_ref, copy_only)

Destination_ref is the range of cells into which you want to fill data. The top, bottom, left, or right end of destination_ref must include all of the cells in the source reference (the current selection).

Copy_only is a number specifying whether to copy cells or perform an AutoFill operation.

Value	Result
0 or omitted	Normal AutoFill
1 or TRUE	Copy cells
2	Copy formats
3	Fill values
4	Increment
5	Increment by day
6	Increment by weekday
7	Increment by month
8	Increment by year
9	Linear trend
10	Growth trend

Related Functions

COPY Copies and pastes data or objects

DATA.SERIES Fills a range of cells with a series of numbers or dates

FILL.DOWN, FILL.LEFT, FILL.RIGHT, FILL.UP

Equivalent to clicking the Down, Left, Right, and Up commands, respectively, on the Fill submenu of the Edit menu.

Syntax

FILL.DOWN( )

FILL.LEFT( )

FILL.RIGHT( )

FILL.UP( )

FILL.DOWN copies the contents and formats of the cells in the top row of a selection into the rest of the rows in the selection.

FILL.LEFT copies the contents and formats of the cells in the right column of a selection into the rest of the columns in the selection.

FILL.RIGHT copies the contents and formats of the cells in the left column of a selection into the rest of the columns in the selection.

FILL.UP copies the contents and formats of the cells in the bottom row of a selection into the rest of the rows in the selection.

Remarks

If you have a multiple selection, each range in the selection is filled separately with the contents of the source range.

Related Functions

COPY	Copies and pastes data or objects
DATA.SERIES	Fills a range of cells with a series of numbers or dates
FILL.AUTO	Copies cells or automatically fills a selection
FORMULA.FILL	Enters a formula in the specified range

FILL.GROUP

Equivalent to choosing the Across Worksheets command from the Fill submenu on the Edit menu. Copies the contents of the active worksheet's selection to the same area on all other worksheets in the group. Use FILL.GROUP to fill a range of cells on all worksheets in a group at once.

Syntax

FILL.GROUP(**type_num**)

FILL.GROUP?(type_num)

Type_num is a number from 1 to 3 that corresponds to the choices in the Fill Across Worksheets dialog box.

Type_num	**Type of information filled**

1	All
2	Contents
3	Formats

Related Functions

NEW   Creates a new workbook

WORKBOOK.SELECT   Selects one or more sheets in a workbook

FILTER

Filters lists of data one column at a time. Only one list can be filtered on any one sheet at a time.

Syntax

FILTER(field_num, criteria1, operation, criteria2)

FILTER?(field_num, criteria1, operation, criteria2)

Field_num   is the number of the field that you want to filter. Fields are numbered from left to right starting with 1.

Criteria1   is a text string specifying criteria for filtering a list, such as ">2". If you want to include all items in the list, omit this argument.

Operation   is a number that specifies how you want criteria2 used with criteria1:

Number	Operation Used
1	AND
2	OR

Criteria2   is a text string specifying criteria for filtering a list, such as ">2". If you include this argument, operation is required.

Remarks

If you omit all arguments, FILTER toggles the display of filter arrows.

Related Function

FILTER.ADVANCED   Lets you set options for filtering a list

FILTER.ADVANCED

Equivalent to choosing the Advanced Filter command from the Filter submenu on the Data menu. Lets you set options for filtering a list.

Syntax

FILTER.ADVANCED(**operation**, **list_ref**, criteria_ref, copy_ref, unique)

FILTER.ADVANCED?(operation, list_ref, criteria_ref, copy_ref, unique)

Operation is a number specifying whether to copy the filter list to a new location. To filter a list without copying, use 1; to copy the filter list to a new location, use 2.

List_ref specifies the location of the list to be filtered. If operation is 1, then list_ref must be on the active sheet.

Criteria_ref is a reference to a range containing criteria for filtering the list. If omitted, uses "All" as the criteria.

Copy_ref is a reference on the active sheet where you want the filtered list copied. Ignored if operation is 1.

Unique is a logical value that specifies whether only unique records are displayed. To display only unique records, use TRUE. To display all records that match the criteria, use FALSE or omit this argument.

Related Function

FILTER Filters lists of data one column at a time

FILTER.SHOW.ALL

Equivalent to choosing the Show All command from the Filter submenu on the Data menu. Displays all items in a filtered list.

Syntax

FILTER.SHOW.ALL()

FIND.FILE

Equivalent to choosing the Find File command from the File menu in Microsoft Excel version 5.0. Lets you search for files based on criteria such as author or creation date.

Syntax

FIND.FILE?( )

Remarks

This function has a dialog-box form only.

FONT

Equivalent to clicking the Font command on the Options menu in Microsoft Excel for the Macintosh version 1.5 or earlier. This function is included only for macro compatibility. Sets the font for the Normal style. Microsoft Excel now uses the FONT.PROPERTIES and DEFINE.STYLE functions. For more information, see FONT.PROPERTIES and DEFINE.STYLE.

Syntax

FONT(**name_text, size_num**)

FONT?(name_text, size_num)

Related Functions

DEFINE.STYLE	Creates or changes a cell style
FONT.PROPERTIES	Sets various font properties

FONT.PROPERTIES

Equivalent to choosing the Cells command from the Format menu. Applies a font and other attributes to the selection. Applies to cells, charts, and text boxes and buttons on worksheets and macro sheets.

Syntax

FONT.PROPERTIES(font, font_style, size, strikethrough, superscript, subscript, outline, shadow, underline, color, normal, background, start_char, char_count)

FONT.PROPERTIES?(font, font_style, size, strikethrough, superscript, subscript, outline, shadow, underline, color, normal, background, start_char, char_count)

Arguments correspond to check boxes or options in the Font tab on the Format Cells dialog box. Arguments that correspond to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the format is not changed.

Font   is the name of the font as it appears on the Font tab. For example, Courier is a font name.

Font_style   is the name of the font style as it appears on the Font tab. For example, Bold Italic is a font style.

Size   is the font size, in points.

Strikethrough   corresponds to the Strikethrough check box.

Superscript   corresponds to the Superscript check box

Subscript   corresponds to the Subscript check box

Outline   corresponds to the Outline check box. Outline fonts are available in Microsoft Excel for the Macintosh. For macro compatibility, this argument is ignored by Microsoft Excel for Windows..

Shadow   corresponds to the Shadow check box. Shadow fonts are available in Microsoft Excel for the Macintosh. For macro compatibility, this argument is ignored by Microsoft Excel for Windows.

Note For macro compatibility with Microsoft Excel for the Macintosh, the presence of the outline and shadow arguments do not prevent the macro from working on Microsoft Excel for Windows, nor does their absence prevent it from working on the Macintosh.

Underline corresponds to the Underline Drop-down box.

Underline	Type applied
0	None
1	Single
2	Double
3	Single Accounting
4	Double Accounting

Color is a number from 0 to 56 corresponding to the colors listed in the Color box; 0 corresponds to automatic color.

Normal corresponds to the Normal Font check box. Applies the default font for your system

Background is a number from 1 to 3 specifying which type of background to apply to text in a chart.

Background	Type of background applied
1	Automatic
2	Transparent
3	Opaque

Start_char specifies the first character to be formatted. If start_char is omitted, it is assumed to be 1 (the first character in the cell or text box).

Char_count specifies how many characters to format. If char_count is omitted, Microsoft Excel formats all characters in the cell or text box starting at start_char.

Remarks

Some extended TrueType styles do not have corresponding arguments to FONT.PROPERTIES. To access an extended TrueType font style, append the style name to the font name in the font argument. For example, the font Taipei can be formatted in an upside-down style by specifying "Taipei Upside-down" as the font argument. For more information about TrueType, see your Microsoft Windows documentation.

Related Functions

ALIGNMENT   Aligns or wraps text in cells

FORMAT.NUMBER   Applies a number format to the selection

FORMAT.TEXT   Formats a worksheet text box or a chart text item

FOPEN

Opens a file with the type of permission specified. Unlike OPEN, FOPEN does not load the file into memory and display it; instead, FOPEN establishes a channel with the file so that you can exchange information with it. If the file is opened successfully, FOPEN returns a file ID number. If it can't open the file, FOPEN returns the #N/A error value. Use the file ID number with other file functions (such as FREAD, FWRITE, and FSIZE) when you want to get information from or send information to the file.

Syntax

FOPEN(**file_text**, access_num)

File_text   is the name of the file you want to open.

Access_num   is a number from 1 to 3 specifying what type of permission to allow to the file:

Access_num	**Type of permission**
1 or omitted	Can read and write to the file (read/write permission)
2	Can read the file, but can't write to the file (read-only permission)
3	Creates a new file with read/write permission

- If the file doesn't exist and access_num is 3, FOPEN creates a new file.
- If the file does exist and access_num is 3, FOPEN replaces the contents of the file with any information you supply using the FWRITE or FWRITELN functions.
- If the file doesn't exist and access_num is 1 or 2, FOPEN returns the #N/A error value.

Remarks

Use FCLOSE to close a file after you finish using it.

Example

The following function opens a file identified as FileName using read-only permission:

```
FOPEN(FileName, 2)
```

Related Functions

FCLOSE Closes a text file

FREAD Reads characters from a text file

FWRITE Writes characters to a text file

OPEN Opens a workbook

FOR

Starts a FOR-NEXT loop. The instructions between FOR and NEXT are repeated until the loop counter reaches a specified value. Use FOR when you need to repeat instructions a specified number of times. Use FOR.CELL when you need to repeat instructions over a range of cells.

Syntax

FOR(**counter_text, start_num, end_num**, step_num)

Counter_text is the name of the loop counter in the form of text.

Start_num is the value initially assigned to counter_text.

End_num is the last value assigned to counter_text.

Step_num is a value added to the loop counter after each iteration. If step_num is omitted, it is assumed to be 1.

Remarks

- Microsoft Excel follows these steps as it executes a FOR-NEXT loop:

Step	Action
1	Sets counter_text to the value start_num.
2	If counter_text is greater than end_num (or less than end_num if step_num is negative), the loop ends, and the macro continues with the function after the NEXT function. If counter_text is less than or equal to end_num (or greater than or equal to end_num if step_num is negative), the macro continues in the loop.
3	Carries out functions up to the following NEXT function. The NEXT function must be below the FOR function and in the same column.
4	Adds step_num to the loop counter.

5 Returns to the FOR function and proceeds as described in step 2.

- You can interrupt a FOR-NEXT loop by using the BREAK function.

Example

The following macro starts a FOR-NEXT loop that is executed once for every open window:

```
FOR("Counter", 1, COLUMNS(WINDOWS()))
```

Related Functions

BREAK Interrupts a FOR-NEXT, FOR.CELL-NEXT, or WHILE-NEXT loop

FOR.CELL Starts a FOR.CELL-NEXT loop

NEXT Ends a FOR-NEXT, FOR.CELL-NEXT, or WHILE-NEXT loop

WHILE Starts a WHILE-NEXT loop

FOR.CELL

Starts a FOR.CELL-NEXT loop. This function is similar to FOR, except that the instructions between FOR.CELL and NEXT are repeated over a range of cells, one cell at a time, and there is no loop counter.

Syntax

FOR.CELL(**ref_name**, area_ref, skip_blanks)

Ref_name is the name in the form of text that Microsoft Excel gives to the one cell in the range that is currently being operated on; ref_name refers to a new cell during each loop.

Area_ref is the range of cells on which you want the FOR.CELL-NEXT loop to operate and can be a multiple selection. If area_ref is omitted, it is assumed to be the current selection.

Skip_blanks is a logical value specifying whether Microsoft Excel skips blank cells as it operates on the cells in area_ref.

Skip_blanks	**Result**
TRUE	Skips blank cells in area_ref
FALSE or omitted	Operates on all cells in area_ref

Remarks

FOR.CELL operates on each cell in a row from left to right one area at a time before moving to the next row in the selection.

Example

The following macro starts a FOR.CELL-NEXT loop and uses the name CurrentCell to refer to the cell in the range that is currently being operated on:

```
FOR.CELL("CurrentCell", SELECTION(), TRUE)
```

Related Functions

BREAK	Interrupts a FOR-NEXT, FOR.CELL-NEXT, or WHILE-NEXT loop
FOR	Starts a FOR-NEXT loop
NEXT	Ends a FOR-NEXT, FOR.CELL-NEXT, or WHILE-NEXT loop
WHILE	Starts a WHILE-NEXT loop

FORMAT.AUTO

Equivalent to clicking the AutoFormat command on the Format menu when a worksheet is active or clicking the AutoFormat button. Formats the selected range of cells from a built-in gallery of formats.

Syntax

FORMAT.AUTO(format_num, number, font, alignment, border, pattern, width)

FORMAT.AUTO?(format_num, number, font, alignment, border, pattern, width)

Format_num is a number from 1 to 17 corresponding to the formats in the Table Format list box in the AutoFormat dialog box.

Format_num	Table Format
0	None
1 or omitted	Classic 1
2	Classic 2
3	Classic 3
4	Accounting 1
5	Accounting 2

6	Accounting 3
7	Colorful 1
8	Colorful 2
9	Colorful 3
10	List 1
11	List 2
12	List 3
13	3D Effects 1
14	3D Effects 2
15	Japan 1 (Far East versions of Microsoft Excel only)
16	Japan 2 (Far East versions of Microsoft Excel only)
17	Accounting 4
18	Simple

The following arguments are logical values corresponding to the Formats To Apply check boxes in the AutoFormat dialog box. If an argument is TRUE or omitted, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box.

Number   corresponds to the Number check box.

Font   corresponds to the Font check box.

Alignment   corresponds to the Alignment check box.

Border   corresponds to the Border check box.

Pattern   corresponds to the Pattern check box.

Width   corresponds to the Column Width/Row Height check box.

Related Functions

ALIGNMENT Aligns or wraps text in cells

BORDER Adds a border to the selected cell or object

FONT.PROPERTIES Applies a font to the selection

FORMAT.NUMBER Applies a number format to the selection

PATTERNS Changes the appearance of the selected object

FORMAT.CHART

Equivalent to choosing the Options button in the Chart Type dialog box, which is available when you choose the Chart Type command from the Format menu when a chart is active. Formats the chart according to the arguments you specify.

Syntax

FORMAT.CHART(layer_num, view, overlap, angle, gap_width, gap_depth, chart_depth, doughnut_size, axis_num, drop, hilo, up_down, series_line, labels, vary)

FORMAT.CHART?(layer_num, view, overlap, angle, gap_width, gap_depth, chart_depth, doughnut_size, axis_num, drop, hilo, up_down, series_line, labels, vary)

Several of the following arguments are logical values corresponding to check boxes in the Options tab of Format (chart type) Group dialog box. If an argument is TRUE, Microsoft Excel selects the corresponding check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the setting is unchanged.

Layer_num is a number specifying which chart you want to change.

View is a number specifying one of the subtypes in the Subtype tab of the Format (type) Group dialog box. The subtype varies depending on the type of chart.

Overlap is a number from -100 to 100 specifying how you want bars or columns to be positioned. It corresponds to the Overlap edit box in the Options tab on the Format Bar Group Dialog box, which appears when you choose the Bar Group from the Format menu. Overlap is ignored if type_num is not 2 or 3 (bar or column chart).

- If overlap is positive, it specifies the percentage of overlap you want for bars or columns. For example, 50 would cause one-half of a bar or column to be covered by an adjacent bar or column. A value of zero prevents bars or columns from overlapping.
- If overlap is negative, then bars or columns are separated by the specified percentage of the maximum available distance between any two bars or columns.
- If overlap is omitted, it is assumed to be 0 (bars or columns do not overlap), or it is unchanged if a value was previously set.

Angle is a number from 0 to 360 specifying the angle of the first pie or doughnut slice (in degrees) if the chart is a pie or doughnut chart. If angle is omitted, it is assumed to be 0, or it is unchanged if a value was previously set.

Gap_width is a number from 0 to 500 specifying the space between bar or column clusters as a percentage of the width of a bar or column. It corresponds to the Gap Width edit box in the Options tab on the Format Bar Group Dialog box, which appears when you choose the Bar Group from the Format menu.

- Gap_width is ignored if type_num is not 2, 3, 8, or 12 (bar or column chart).
- If Gap_width is omitted, it is assumed to be 50, or it is unchanged if a value was previously set.

The next two arguments are for 3-D charts only, and correspond to check boxes in the Options tab of Format (chart type) Group dialog box.

Gap_depth is a number from 0 to 500 specifying the depth of the gap in front of and behind a bar, column, area, or line as a percentage of the depth of the bar, column, area, or line.

- Gap_depth is ignored if the chart is a pie chart or if it is not a 3-D chart.
- If gap_depth is omitted and the chart is a 3-D chart, gap_depth is assumed to be 50, or it is unchanged if a value was previously set. If gap_depth is omitted and the view is side-by-side, stacked, or stacked 100%, gap_depth is assumed to be 0, or it is unchanged if a value was previously set.

Chart_depth is a number from 20 to 2000 specifying the visual depth of the chart as a percentage of the width of the chart.

- Chart_depth is ignored if the chart is not a 3-D chart.
- If Chart_depth is omitted, it is assumed to be 100, or it is unchanged if a value was previously set.

Doughnut_size specifies the size of the hole in a doughnut chart. Can be a value from 10% - 90%. Default is 50%.

Axis_num is a number specifying whether to plot the chart on the primary axis or the secondary axis.

Drop corresponds to the Drop Lines check box. Drop is available only for area and line charts.

Hilo corresponds to the Hi-Lo Lines check box. Hilo is available only for 2-D line charts.

The next four arguments are logical values corresponding to check boxes in the Options tab of the Format (chart type) Group dialog box. If an argument is TRUE, Microsoft Excel selects the corresponding check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the setting is unchanged.

Up_down corresponds to the Up/Down Bars check box. Up_down is available only for 2-D line charts.

Series_line corresponds to the Series Lines check box. Series_line is available only for 2-D stacked bar and column charts.

Labels corresponds to the Radar Axis Labels check box. Labels is available only for radar charts.

Vary corresponds to the Vary Colors By Point check box. Vary applies only to charts with one data series and is not available for area charts.

Related Functions

FORMAT.MAIN Formats a chart according to the arguments you specify

FORMAT.OVERLAY Formats an overlay chart

FORMAT.CHARTTYPE

Changes the chart type for a selected data series, a group of data series, or an entire chart.

Syntax

FORMAT.CHARTTYPE(**apply_to**, group_num, dimension, **type_num**)

FORMAT.CHARTTYPE?(apply_to, group_num, dimension, type_num)

Apply_to is a number from 1 to 3 specifying what part of a chart the new chart type effects.

Value	Part of chart
1	Selected data series
2	Group of data series
3	Entire chart

Group_num corresponds to the number of the group you want to change as listed in the Group list box of the Chart Type dialog box, which appears when you click Chart Type from the Format menu while a chart is active. Groups are numbered starting with 1 for the group at the top of the list. This argument is required if apply_to equals 2; otherwise it is ignored.

Dimension specifies whether to apply a 2-D or 3-D chart type. Use 1 for a 2-D chart type or 2 for a 3-D chart type. If omitted, uses the same dimension as the series, group, or chart to be changed.

Type_num specifies the chart type to apply. Meaning of type_num varies depending on the value of dimension:

Type_num	Chart type if dimension is 1
1	Area or 3-D area
2	Bar or 3-D bar
3	Column or 3-D column
4	Line or 3-D line

5	Pie or 3-D pie
6	Doughnut or 3-D surface
7	Radar
8	XY (scatter)

Related Function

FORMAT.CHART Formats the selected chart

FORMAT.FONT

Equivalent to choosing the Cells command from the Format menu, and then selecting Font tab from the Format Cells dialog box. This function is included for compatibility with Microsoft Excel version 4.0. Use FONT.PROPERTIES to set various font properties. FORMAT.FONT has three syntax forms. Syntax 1 is for cells; syntax 2 is for text boxes and buttons; syntax 3 is used with all chart items (axes, labels, text, and so on).

Syntax 1

Cells

FORMAT.FONT(name_text, size_num, bold, italic, underline, strike, color, outline, shadow)

FORMAT.FONT?(name_text, size_num, bold, italic, underline, strike, color, outline, shadow)

Syntax 2

Text boxes and buttons on worksheets and macro sheets

FORMAT.FONT(name_text, size_num, bold, italic, underline, strike, color, outline, shadow, object_id_text, start_num, char_num)

FORMAT.FONT?(name_text, size_num, bold, italic, underline, strike, color, outline, shadow, object_id_text, start_num, char_num)

Syntax 3

Chart items including unattached chart text

FORMAT.FONT(color, backgd, apply, name_text, size_num, bold, italic, underline, strike, outline, shadow, object_id_text, start_num, char_num)

FORMAT.FONT?(color, backgd, apply, name_text, size_num, bold, italic, underline, strike, outline, shadow, object_id_text, start_num, char_num)

Arguments correspond to check boxes and list box items in the Font tab on the Format Cells dialog box. Arguments that correspond to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the format is not changed.

Name_text is the name of the font as it appears in the Font tab. For example, Courier is a font name.

Size_num is the font size, in points.

Bold corresponds to the Bold item in the Font Style list box. Makes the selection bold, if applicable.

Italic corresponds to the Italic item in the Font Style list box. Makes the selection italic, if applicable.

Underline corresponds to the Underline check box.

Strike corresponds to the Strikethrough check box.

Color is a number from 0 to 56 corresponding to the colors in the Font tab; 0 corresponds to automatic color.

Outline corresponds to the Outline check box. Outline fonts are available in Microsoft Excel for the Macintosh. For macro compatibility, this argument is ignored by Microsoft Excel for Windows.

Shadow corresponds to the Shadow check box. Shadow fonts are available in Microsoft Excel for the Macintosh. For macro compatibility, this argument is ignored by Microsoft Excel for Windows.

Note For macro compatibility with Microsoft Excel for the Macintosh, the presence of the outline and shadow arguments do not prevent the macro from working on Microsoft Excel for Windows, nor does their absence prevent it from working on the Macintosh.

Object_id_text identifies the text box you want to format (for example, "Text 1", "Text 2", and so on). You can also use the object number alone without the text identifier. For compatibility with earlier versions of Microsoft Excel. This argument is ignored in Microsoft Excel version 5.0 or later.

Start_num specifies the first character to be formatted. If start_num is omitted, it is assumed to be 1 (the first character in the text box).

Char_num specifies how many characters to format. If char_num is omitted, Microsoft Excel formats all characters in the text box starting at start_num.

Backgd is a number from 1 to 3 specifying which type of background to apply to text in a chart.

Backgd	Type of background applied
1	Automatic
2	Transparent
3	Opaque

Apply corresponds to the Apply To All check box. This argument applies to data labels only.

Remarks

Some extended TrueType styles do not have corresponding arguments to FORMAT.FONT. To access an extended TrueType font style, append the style name to the font name in name_text. For example, the font Taipei can be formatted in an upside-down style by specifying "Taipei Upside-down" as the name_text argument. For more information about TrueType, see your Microsoft Windows documentation.

Related Functions

ALIGNMENT   Aligns or wraps text in cells

FONT.PROPERTIES   Sets various font attributes

FORMAT.NUMBER   Applies a number format to the selection

FORMAT.TEXT   Formats a worksheet text box or a chart text item

FORMAT.LEGEND

Equivalent to clicking the Selected Legend command on the Format menu when a chart is active. Determines the position and orientation of the legend on a chart and returns TRUE; returns an error message if the legend is not already selected.

Syntax

FORMAT.LEGEND(**position_num**)

FORMAT.LEGEND?(position_num)

Position_num   is a number from 1 to 5 specifying the position of the legend.

Position_num	**Position of legend**
1	Bottom
2	Corner
3	Top
4	Right
5	Left

Related Functions

FORMAT.MOVE   Moves the selected object

FORMAT.SIZE   Sizes an object

LEGEND   Adds or deletes a chart legend

FORMAT.MAIN

Equivalent to clicking the Main Chart command on the Format menu in Microsoft Excel version 4.0. Formats a chart according to the arguments you specify. This function is included for compatibility with Microsoft Excel version 4.0. In Microsoft Excel version 5.0 or later, this is equivalent to clicking the Chart Type command on the Format menu. You can also use the FORMAT.CHART function.

Syntax

FORMAT.MAIN(**type_num**, view, overlap, gap_width, vary, drop, hilo, angle, gap_depth, chart_depth, up_down, series_line, labels, doughnut_size)

FORMAT.MAIN?(type_num, view, overlap, gap_width, vary, drop, hilo, angle, gap_depth, chart_depth, up_down, series_line, labels, doughnut_size)

Type_num is a number specifying the type of chart.

Type_num	Chart
1	Area
2	Bar
3	Column
4	Line
5	Pie
6	XY (Scatter)
7	3-D Area
8	3-D Column
9	3-D Line
10	3-D Pie
11	Radar
12	3-D Bar

13	3-D Surface
14	Doughnut

View is a number specifying one of the views in the Data View box in the Main Chart dialog box. The view varies depending on the type of chart.

Overlap is a number from -100 to 100 specifying how you want bars or columns to be positioned. It corresponds to the Overlap box in the Main Chart dialog box. Overlap is ignored if type_num is not 2 or 3 (bar or column chart).

- If overlap is positive, it specifies the percentage of overlap you want for bars or columns. For example, 50 would cause one-half of a bar or column to be covered by an adjacent bar or column. A value of zero prevents bars or columns from overlapping.
- If overlap is negative, then bars or columns are separated by the specified percentage of the maximum available distance between any two bars or columns.
- If overlap is omitted, it is assumed to be 0 (bars or columns do not overlap), or it is unchanged if a value was previously set.

Gap_width is a number from 0 to 500 specifying the space between bar or column clusters as a percentage of the width of a bar or column. It corresponds to the Gap Width box in the Main Chart dialog box.

- Gap_width is ignored if type_num is not 2, 3, 8, or 12 (bar or column chart).
- If gap_width is omitted, it is assumed to be 50, or it is unchanged if a value was previously set.

Several of the following arguments are logical values corresponding to check boxes in the Main Chart dialog box. If an argument is TRUE, Microsoft Excel selects the corresponding check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the setting is unchanged.

Vary corresponds to the Vary By Categories check box. Vary applies only to charts with one data series and is not available for area charts.

Drop corresponds to the Drop Lines check box. Drop is available only for area and line charts.

Hilo corresponds to the Hi-Lo Lines check box. Hilo is available only for line charts.

Angle is a number from 0 to 360 specifying the angle of the first pie slice (in degrees) if the chart is a pie chart. If angle is omitted, it is assumed to be 0, or it is unchanged if a value was previously set.

The next two arguments are for 3-D charts only.

Gap_depth is a number from 0 to 500 specifying the depth of the gap in front of and behind a bar, column, area, or line as a percentage of the depth of the bar, column, area, or line.

- Gap_depth is ignored if the chart is a pie chart or if it is not a 3-D chart.

- If gap_depth is omitted and the chart is a 3-D chart, gap_depth is assumed to be 50, or it is unchanged if a value was previously set. If gap_depth is omitted and the view is side-by-side, stacked, or stacked 100%, gap_depth is assumed to be 0, or it is unchanged if a value was previously set.

Chart_depth   is a number from 20 to 2000 specifying the visual depth of the chart as a percentage of the width of the chart. Chart_depth corresponds to the Chart Depth box in the Main Chart dialog box.

- Chart_depth is ignored if the chart is not a 3-D chart.
- If chart_depth is omitted, it is assumed to be 100, or it is unchanged if a value was previously set.

The next three arguments are logical values corresponding to check boxes in the Main Chart dialog box. If an argument is TRUE, Microsoft Excel selects the corresponding check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the setting is unchanged. The final argument is for compatibility with Microsoft Excel version 4.0.

Up_down   corresponds to the Up/Down Bars check box. Up_down is available only for line charts.

Series_line   corresponds to the Series Lines check box. Series_line is available only for stacked bar and column charts.

Labels   corresponds to the Radar Axis Labels check box. Labels is available only for radar charts.

Doughnut_size   specifies the size of the hole in a doughnut chart. Can be a value from 10% - 90%. Default is 50%

Related Functions

FORMAT.CHART   Formats a chart

FORMAT.OVERLAY   Formats an overlay chart

FORMAT.MOVE

Equivalent to moving an object with the mouse. Moves the selected object to the specified position and, if successful, returns TRUE. If the selected object cannot be moved, FORMAT.MOVE returns FALSE. There are three syntax forms of this function. Use syntax 1 to move worksheet objects. Use syntax 2 to move chart items. Use syntax 3 to move pie-chart and doughnut-chart items. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax 1	Moves worksheet items
Syntax 2	Moves chart items
Syntax 3	Moves pie-chart and doughnut-chart items

FORMAT.MOVE SYNTAX 1

Equivalent to moving an object with the mouse. Moves the selected object to the specified position and, if successful, returns TRUE. If the selected object cannot be moved, FORMAT.MOVE returns FALSE. There are three syntax forms of this function. Use syntax 1 to move worksheet objects. Use syntax 2 to move chart items. Use syntax 3 to move pie-chart and doughnut-chart items. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax

FORMAT.MOVE(**x_offset**, **y_offset**, reference)

FORMAT.MOVE?(x_offset, y_offset, reference)

X_offset specifies the horizontal position to which you want to move the object and is measured in points from the upper-left corner of the object to the upper-left corner of the cell specified by reference. A point is 1/72nd of an inch.

Y_offset specifies the vertical position to which you want to move the object and is measured in points from the upper-left corner of the object to the upper-left corner of the cell specified by reference.

Reference specifies which cell or range of cells to place the object in relation to.

- If reference is a range of cells, only the upper-left cell is used.
- If reference is omitted, it is assumed to be cell A1.

Remarks

The position of an object is based on its upper-left corner. For ovals and arcs, the position is based on the upper-left corner of the bounding rectangle of the object.

Example

The following macro formula moves an object on the active worksheet so that it is 10 points horizontally offset and 15 points vertically offset from cell D4:

```
FORMAT.MOVE(10, 15, !$D$4)
```

Related Functions

CREATE.OBJECT	Creates an object
FORMAT.SIZE	Sizes an object
WINDOW.MOVE	Moves a window
Syntax 2	Moves chart items
Syntax 3	Moves pie-chart and doughnut-chart items

FORMAT.MOVE SYNTAX 2

Equivalent to moving an object with the mouse. Moves the base of the selected object to the specified position and, if successful, returns TRUE. If the selected object cannot be moved, FORMAT.MOVE returns FALSE. There are three syntax forms of this function. Use syntax 3 to move pie-chart and doughnut-chart items. Use syntax 1 to move worksheet objects. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax

FORMAT.MOVE(**x_pos**, **y_pos**)

FORMAT.MOVE?(x_pos, y_pos)

X_pos specifies the horizontal position to which you want to move the object and is measured in points from the base of the object to the lower-left corner of the window. A point is 1/72nd of an inch.

Y_pos specifies the vertical position to which you want to move the object and is measured in points from the base of the object to the lower-left corner of the window.

Remarks

- The base of a text label on a chart is the lower-left corner of the text rectangle.
- The base of an arrow is the end without the arrowhead.
- The base of a pie slice is the point.

Example

On a chart, the following macro formula moves the base of the selected chart object 10 points to the right of and 20 points above the lower-left corner of the window:

```
FORMAT.MOVE(10, 20)
```

Related Functions

FORMAT.SIZE	Sizes an object
WINDOW.MOVE	Moves a window
Syntax 1	Moves worksheet items
Syntax 3	Moves pie-chart and doughnut-chart items

FORMAT.MOVE SYNTAX 3

Equivalent to exploding by moving a pie-chart or doughnut-chart slice with the mouse. Sets the percentage of pie-chart or doughnut-chart slice explosion, and, if successful, returns TRUE. If the selected object cannot be exploded, returns FALSE. There are three syntax forms of this function. Use syntax 1 to move worksheet items. Use syntax 2 to move chart items. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax

FORMAT.MOVE(**explosion_num**)

Explosion_num is a number specifying the explosion percentage for the selected pie slice or the entire chart (if the series is selected). Zero is no explosion (the tip of the slice is in the center of the pie).

Related Functions

FORMAT.SIZE Sizes an object

Syntax 1 Moves worksheet items

Syntax 2 Moves chart items

WINDOW.MOVE Moves a window

FORMAT.NUMBER

Equivalent to choosing the Number tab in the Format Cells dialog box, which appears when you choose Cells from the Format menu. Formats numbers, dates, and times in the selected cells, data labels, and axis labels on charts. Use FORMAT.NUMBER to apply built-in formats or to create and apply custom formats.

Syntax

FORMAT.NUMBER(**format_text**)

FORMAT.NUMBER?(format_text)

Format_text is a format string, such as "#, ##0.00", specifying which format to apply to the selection.

Related Functions

DELETE.FORMAT Deletes the specified custom number format

FONT.PROPERTIES Applies a font to the selection

FORMAT.TEXT Formats a sheet text box or a chart text item

FORMAT.OVERLAY

Equivalent to clicking the Overlay command on the Format menu in Microsoft Excel version 4.0. Formats the overlay chart according to the arguments you specify.

Syntax

FORMAT.OVERLAY(**type_num**, view, overlap, gap_width, vary, drop, hilo, angle, series_dist, series_num, up_down, series_line, labels)

FORMAT.OVERLAY?(type_num, view, overlap, gap_width, vary, drop, hilo, angle, series_dist, series_num, up_down, series_line, labels)

Type_num is a number specifying the type of chart.

Type_num	Chart
1	Area
2	Bar
3	Column
4	Line
5	Pie
6	XY (Scatter)
11	Radar
14	Doughnut

View is a number specifying one of the views in the Data View box in the Overlay dialog box. The view varies depending on the type of chart.

Overlap is a number from -100 to 100 specifying how you want bars or columns to be positioned. It corresponds to the Overlap box in the Overlay dialog box. Overlap is ignored if type_num is not 2 or 3 (bar or column chart).

- If overlap is positive, it specifies the percentage of overlap you want for bars or columns. For example, 50 would cause one-half of a bar or column to be covered by an adjacent bar or column.
- If overlap is negative, then bars or columns are separated by the specified percentage of the maximum available distance between any two bars or columns.
- If overlap is omitted, it is assumed to be 0 (bars or columns do not overlap), or it is unchanged if a value was previously set.

Gap_width is a number from 0 to 500 specifying the space between bar or column clusters as a percentage of the width of a bar or column.

- Gap_width is ignored if type_num is not 2 or 3 (bar or column chart).
- If gap_width is omitted, it is assumed to be 50, or it is unchanged if a value was previously set.

Several of the following arguments are logical values corresponding to check boxes in the Overlay dialog box. If an argument is TRUE, Microsoft Excel selects the corresponding check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the setting is unchanged.

Vary   corresponds to the Vary By Categories check box. Vary is not available for area charts.

Drop   corresponds to the Drop Lines check box. Drop is available only for area and line charts.

Hilo   corresponds to the Hi-Lo Lines check box. Hilo is available only for line charts.

Angle   is a number from 0 to 360 specifying the angle of the first pie slice (in degrees) if the chart is a pie chart. If angle is omitted, it is assumed to be 0, or it is unchanged if a value was previously set.

Series_dist   is the number 1 or 2 and specifies automatic or manual series distribution.

- If series_dist is 1 or omitted, Microsoft Excel uses automatic series distribution.
- If series_dist is 2, Microsoft Excel uses manual series distribution, and you must specify which series is first in the distribution by using the series_num argument.

Series_num   is the number of the first series in the overlay chart and corresponds to the First Overlay Series box in the Overlay dialog box. If series_dist is 1 (automatic series distribution), this argument is ignored.

Up_down   corresponds to the Up/Down Bars check box. Up_down is available only for line charts.

Series_line   corresponds to the Series Lines check box. Series_line is available only for stacked bar and column charts.

Labels   corresponds to the Radar Axis Labels check box. Labels is available only for radar charts.

Related Functions

DELETE.OVERLAY   Deletes the overlay on a chart

FORMAT.CHART   Formats a chart

FORMAT.SHAPE

Equivalent to clicking the reshape button on the Drawing toolbar and then inserting, moving, or deleting vertices of the selected polygon. A vertex is a point defined by a pair of coordinates in one row of the array that defines the polygon. The array is created by CREATE.OBJECT and EXTEND.POLYGON functions.

Syntax

FORMAT.SHAPE(**vertex_num, insert**, reference, x_offset, y_offset)

Vertex_num   is a number corresponding to the vertex you want to insert, move, or delete.

Insert   is a logical value specifying whether to insert a vertex, or move or delete a vertex.

- If insert is TRUE, Microsoft Excel inserts a vertex between the vertices vertex_num and vertex_num-1. The number of the new vertex then becomes vertex_num. The number of the vertex previously identified by vertex_num becomes vertex_num+1, and so on.

- If insert is FALSE, Microsoft Excel deletes the vertex (if the remaining arguments are omitted) or moves the vertex to the position specified by the remaining arguments.

Reference is the reference from which the vertex you are inserting or moving is measured; that is, the cell or range of cells to use as the basis for the x and y offsets.

- If reference is a range of cells, only the upper-left cell is used.

- If reference is omitted, the vertex is measured from the upper-left corner of the polygon's bounding rectangle.

X_offset is the horizontal distance from the upper-left corner of reference to the vertex. X_offset is measured in points. A point is 1/72nd of an inch. If reference is omitted, x_offset specifies the horizontal distance from the upper-left corner of the polygon bounding rectangle.

Y_offset is the vertical distance from the upper-left corner of reference to the vertex. Y_offset is measured in points. If reference is omitted, y_offset specifies the vertical distance from the upper-left corner of the polygon bounding rectangle.

Remarks

You cannot delete a vertex if only two vertices remain.

Examples

The following macro formula deletes the second vertex of the selected polygon:

```
FORMAT.SHAPE(2, FALSE)
```

The following macro formula moves the thirteenth vertex 6 points to the right and 4 points below the upper-left corner of cell B5 on the active worksheet:

```
FORMAT.SHAPE(13, FALSE, !$B$5, 6, 4)
```

The following macro formula inserts a new vertex between vertices 2 and 3. The new vertex is 60 points to the right and 75 points below the upper-left corner of the polygon's bounding rectangle:

```
FORMAT.SHAPE(3, TRUE, , 60, 75)
```

Related Functions

CREATE.OBJECT	Creates an object
EXTEND.POLYGON	Adds vertices to a polygon

FORMAT.SIZE

Equivalent to sizing an object with the mouse. Sizes the selected object and returns TRUE. If the selected chart object cannot be sized, FORMAT.SIZE returns FALSE. There are two syntax forms of this function. Use syntax 1 to size worksheet objects and chart items absolutely. Use syntax 2 relative to a cell or range of cells to size only worksheet objects. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax 1   Sizes worksheet objects and chart items

Syntax 2   Sizes worksheet objects relative to a cell or range

FORMAT.SIZE SYNTAX 1

Equivalent to sizing an object with the mouse. Sizes the selected object and returns TRUE. If the selected chart object cannot be sized, FORMAT.SIZE returns FALSE. There are two syntax forms of this function. Use syntax 1 to size worksheet objects and chart items absolutely. Use syntax 2 relative to a cell or range of cells to size only worksheet objects. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax

FORMAT.SIZE(width, height)

FORMAT.SIZE?(width, height)

Width   specifies the width of the selected object, measured in points. A point is 1/72nd of an inch.

Height   specifies the height of the selected object, measured in points.

You do not always have to use both arguments. For example, if you specify height and not width, the height changes but the width does not.

Remarks

- The base of a text label on a chart is the lower-left corner of the text rectangle.
- The base of an arrow is the end without the arrowhead.

Related Functions

FORMAT.MOVE   Moves the selected object

SIZE   Changes the size of a window

Syntax 2   Sizes worksheet objects relative to a cell or range

FORMAT.SIZE SYNTAX 2

Equivalent to sizing an object with the mouse. Sizes the selected worksheet object and returns TRUE. If the selected object cannot be sized, FORMAT.SIZE returns FALSE. There are two syntax forms of this function. Use syntax 2 to size worksheet objects relative to a cell or range of cells. Use syntax 1 to size worksheet objects and chart items. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax

FORMAT.SIZE(x_off, y_off, reference)

FORMAT.SIZE?(x_off, y_off, reference)

X_off specifies the width of the selected object and is measured in points from the lower-right corner of the object to the upper-left corner of reference. A point is 1/72nd of an inch. If omitted, x_off is assumed to be 0. If reference is omitted, x_off specifies the horizontal size.

Y_off specifies the height of the selected object and is measured in points from the lower-right corner of the object to the upper-left corner of reference. If omitted, y_off is assumed to be 0. If reference is omitted, y_off specifies the vertical size.

Reference specifies the cell or range of cells to use as the basis for the offset and for sizing. If reference is a range of cells, only the upper-left cell in the range is used.

Related Functions

FORMAT.MOVE Moves the selected object

SIZE Changes the size of a window

Syntax 1 Sizes worksheet objects and chart items

FORMAT.TEXT

Formats the selected worksheet text box or button or any text item on a chart.

Syntax

FORMAT.TEXT(x_align, y_align, orient_num, auto_text, auto_size, show_key, show_value, add_indent)

FORMAT.TEXT?(x_align, y_align, orient_num, auto_text, auto_size, show_key, show_value, add_indent)

Arguments correspond to check boxes or options in the various tabs on Format Object dialog box. Arguments that correspond to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box; if omitted, the current setting is used.

X_align is a number from 1 to 4 specifying the horizontal alignment of the text.

X_align	Horizontal alignment
1	Left
2	Center
3	Right
4	Justify

Y_align is a number from 1 to 4 specifying the vertical alignment of the text.

Y_align	Vertical alignment
1	Top
2	Center
3	Bottom
4	Justify

Orient_num is a number from 0 to 3 specifying the orientation of the text.

Orient_num	Text orientation
0	Horizontal
1	Vertical
2	Upward
3	Downward

Auto_text corresponds to the Automatic Text check box. If the selected text was created with the Data Labels command from the Insert menu and later edited, this option restores the original text. Auto_text is ignored for text boxes on worksheets and macro sheets.

Auto_size corresponds to the Automatic Size check box. If you have changed the size of the border around the selected text, this option restores the border to automatic size. Automatic size makes the border fit exactly around the text no matter how you change the text.

Show_key corresponds to the Show Legend Key Next to Label check box in the Data Labels dialog box. This argument applies only if the selected text is an attached data label on a chart.

Show_value corresponds to the Show Value option button in the Format Data Labels dialog box. This argument applies only if the selected text is an attached data label on a chart.

The following list summarizes which arguments apply to each type of text item.

Add_indent This argument is for only Far East versions of Microsoft Excel.

Text item	Arguments that apply
Worksheet text box or button	X_align, y_align, orient_num, auto_size
Attached data label	All arguments
Unattached text label	X_align, y_align, orient_num, auto_size
Tickmark label	Orient_num

Related Functions

CREATE.OBJECT	Creates an object
FONT.PROPERTIES	Applies a font to the selection
FORMULA	Enters values into a cell or range or onto a chart

FORMULA

Enters a formula in the active cell or in a reference. There are two syntax forms of this function. Use syntax 1 to enter numbers, text, references, and formulas in a worksheet. Although syntax 1 can also be used to enter values on a macro sheet, you will not generally use FORMULA for this purpose. Use syntax 2 to enter a formula in a chart. For information about setting values on a macro sheet, see "Remarks" in the following topics.

Syntax 1	Enters numbers, text, references, and formulas in a worksheet
Syntax 2	Enters formulas in a chart

FORMULA SYNTAX 1

Enters a formula in the active cell or in a reference. If the active sheet is a worksheet, using FORMULA is equivalent to entering formula_text in the cell specified by reference. Formula_text is entered just as if you typed it in the formula bar.

There are two syntax forms of this function. Use syntax 1 to enter numbers, text, references, and formulas in a worksheet. Although syntax 1 can also be used to enter values on a macro sheet, you will not generally use FORMULA for this purpose. Use syntax 2 to enter a formula in a chart. For information about setting values on a macro sheet, see "Remarks" later in this topic.

Syntax

FORMULA(**formula_text**, reference)

Formula_text can be text, a number, a reference, or a formula in the form of text, or a reference to a cell containing any of the above.

- If formula_text contains references, they must be R1C1-style references, such as "=RC[1]*(1+R1C1)". If you are recording a macro when you enter a formula, Microsoft Excel converts A1-style references to R1C1-style references. For example, if you enter the formula =B2*(1+$A$1) in cell C2 while recording, Microsoft Excel records that action as =FORMULA("=RC[-1]*(1+R1C1)").

- If formula_text is a formula, the formula is entered. Text arguments must be surrounded by double sets of quotation marks. For example, to enter the formula =IF($A$1="Hello World", 1, 0) in the active cell with the FORMULA function, you would use the formula FORMULA("=IF(R1C1=""Hello World"", 1, 0)")

- If formula_text is a number, text, or logical value, the value is entered as a constant.

Reference specifies where formula_text is to be entered. It can be a reference to a cell in the active workbook or an external reference to a workbook. If reference is omitted, formula_text is entered in the active cell.

Remarks

Consider the following guidelines as you choose a function to set values on a worksheet or macro sheet:

- Use FORMULA to enter formulas and change values in a worksheet cell.

- SET.VALUE changes values on the macro sheet. Use SET.VALUE to assign initial values to a reference and to store values during the calculation of the macro.

- SET.NAME creates names on the macro sheet. Use SET.NAME to create a name and immediately assign a value to the name.

Examples

If the active sheet is a worksheet, the following macro formula enters the number constant 523 in the active cell:

```
FORMULA(523)
```

If the active sheet is a worksheet, the following macro formula enters the result of the INPUT function in cell A5:

```
FORMULA(INPUT("Enter a formula:", 0), !$A$5)
```

If you're using R1C1-style references and the active sheet is a worksheet, the following macro formula enters the formula =RC[-1]*(1+R1C1) in the active cell:

```
FORMULA("=RC[-1]*(1+R1C1)")
```

If the active sheet is a worksheet, the following macro formulas enter the number 1000 in the cell two rows down and three columns right from the active cell. The R1C1-style formula is shorter, but the OFFSET method may provide faster performance in larger macro sheets.

```
FORMULA(1000, OFFSET(ACTIVE.CELL(), 2, 3))
FORMULA(1000, "R[2]C[3]")
```

The following macro formula enters the phrase "Year to Date" in cell B4 on the sheet named SALES 1993:

```
FORMULA("Year to Date", 'SALES 1993'!B4)
```

Related Functions

FORMULA.ARRAY	Enters an array
FORMULA.FILL	Enters a formula in the specified range
SET.VALUE	Sets the value of a cell on a macro sheet
FORMULA Syntax 2	Enters formulas in a chart

FORMULA SYNTAX 2

Enters a text label or SERIES formula in a chart. To enter formulas on a worksheet or macro sheet, use syntax 1 of this function.

Syntax

FORMULA(**formula_text**)

Formula_text is the text label or SERIES formula you want to enter into the chart.

If	**Then**
Formula_text can be treated as a text label and the current selection is a text label	The selected text label is replaced with formula_text.
Formula_text can be treated as a text label and there is no current selection or the current selection is not a text label	Formula_text creates a new unattached text label.
Formula_text can be treated as a SERIES formula and the current selection is a SERIES formula	The selected SERIES formula is replaced with formula_text.
Formula_text can be treated as a SERIES formula and the current selection is not a SERIES formula	Formula_text creates a new SERIES formula.

Remarks

You would normally use the EDIT.SERIES function to create or edit a chart series. For more information, see EDIT.SERIES.

Example

The following macro formula enters a SERIES formula on the chart. If the current selection is a SERIES formula, it is replaced:

```
FORMULA("=SERIES(""Title"", , {1, 2, 3}, 1)")
```

Related Functions

EDIT.SERIES   Creates or changes a chart series

FORMULA, Syntax 1   Enters numbers, text, references, and formulas in a worksheet

FORMULA.ARRAY

Enters a formula as an array formula in the range specified or in the current selection. Equivalent to entering an array formula while pressing CTRL+SHIFT+ENTER in Microsoft Excel for Windows or COMMAND+ENTER in Microsoft Excel for the Macintosh.

Syntax

FORMULA.ARRAY(**formula_text**, reference)

Formula_text   is the text you want to enter in the array. For more information on formula_text, see the first form of FORMULA.

Reference   specifies where formula_text is entered. It can be a reference to a cell on the active worksheet or an external reference to a named workbook. Reference must be a R1C1-style reference in text form. If reference is omitted, formula_text is entered in the active cell.

Examples

If the selection is D25:E25, the following macro formula enters the array formula {=D22:E22+D23:E23} in the range D25:E25:

```
FORMULA.ARRAY("=R[-3]C:R[-3]C[1]+R[-2]C:R[-2]C[1]")
```

Regardless of the selection, the following macro formula enters the array formula {=D22:E22+D23:E23} in the range D25:E25:

```
FORMULA.ARRAY("=R[-3]C:R[-3]C[1]+R[-2]C:R[-2]C[1]", "R25C4:R25C5")
```

To use FORMULA.ARRAY to put an array in a specific workbook, specify the name of the workbook as an external reference in the reference argument. Using "[SALES.XLS]North!R25C3:R25C4" as the reference argument in the preceding example would enter the array in cells C25:D25 on the worksheet named North in the workbook SALES.XLS. Using "SALES!R25C3:R25C4" as the reference argument would enter the array in the same cells in the worksheet named SALES.

Related Functions

FORMULA   Enters values into a cell or range or onto a chart

FORMULA.FILL   Enters a formula in the specified range

FORMULA.CONVERT

Changes the style and type of references in a formula between A1 and R1C1 and between relative and absolute. Use FORMULA.CONVERT to convert references of one style or type to another style or type.

Syntax

FORMULA.CONVERT(**formula_text, from_a1**, to_a1, to_ref_type, rel_to_ref)

Formula_text is the formula, given as text, containing the references you want to change. Formula_text must be a valid formula, and an equal sign must be included.

From_a1 is a logical value specifying whether the references in formula_text are in A1 or R1C1 style. If from_a1 is TRUE, references are in A1 style; if FALSE, references are in R1C1 style.

To_a1 is a logical value specifying the form for the references FORMULA.CONVERT returns. If to_a1 is TRUE, references are returned in A1 style; if FALSE, references are returned in R1C1 style. If to_a1 is omitted, the reference style is not changed.

To_ref_type is a number from 1 to 4 specifying the reference type of the returned formula. If to_ref_type is omitted, the reference type is not changed.

To_ref_type	**Reference type returned**
1	Absolute
2	Absolute row, relative column
3	Relative row, absolute column
4	Relative

Rel_to_ref is an absolute reference that specifies what cell the relative references are or should be relative to.

Examples

Use FORMULA.CONVERT to convert relative references entered by the user in an INPUT function or custom dialog box into absolute references. The following macro formula converts the given formula to an absolute, R1C1-style reference:

`FORMULA.CONVERT("=A1:A10", TRUE, FALSE, 1)` equals "=R1C1:R10C1"

The following macro formula converts the references in the given formula to relative, A1-style references:

`FORMULA.CONVERT("=SUM(R10C2:R15C2)", FALSE, TRUE, 4)` equals "=SUM(B10:B15)"

Tip To put the converted formula into a cell or range of cells, use the FORMULA.CONVERT function as the formula_text argument to the FORMULA function.

Related Functions

ABSREF Returns the absolute reference of a range of cells to another range

FORMULA Enters values into a cell or range or onto a chart

RELREF Returns a relative reference

FORMULA.FILL

Enters a formula in the range specified or in the current selection. Equivalent to entering a formula in a range of cells while pressing CTRL+ENTER in Microsoft Excel for Windows or OPTION+ENTER in Microsoft Excel for the Macintosh.

Syntax

FORMULA.FILL(**formula_text**, reference)

Formula_text is the text with which you want to fill the range. For more information on formula_text, see FORMULA.

Reference specifies where formula_text is entered. It can be a reference to a range in the active worksheet or an external reference to a named workbook. If omitted, formula_text is entered in the current selection.

Related Functions

DATA.SERIES Fills a range of cells with a series of numbers or dates

FORMULA Enters values into a cell or range or onto a chart

FORMULA.ARRAY Enters an array

FORMULA.FIND

Equivalent to clicking the Find command on the Edit menu. Selects the next or previous cell containing the specified text and returns TRUE. If a matching cell is not found, FORMULA.FIND returns FALSE and displays a message.

Syntax

FORMULA.FIND(**text, in_num, at_num, by_num**, dir_num, match_case)

FORMULA.FIND?(text, in_num, at_num, by_num, dir_num, match_case)

Text is the text you want to find. Text corresponds to the Find What box in the Find dialog box.

In_num is a number from 1 to 3 specifying where to search.

In_num	Searches
1	Formulas
2	Values

3	Notes

At_num is the number 1 or 2 and specifies whether to find cells containing only text or also cells containing text within a longer string of characters.

At_num	Searches for text as
1	A whole string (the only value in the cell)
2	Either a whole string or part of a longer string

By_num is the number 1 or 2 and specifies whether to search by rows or by columns.

By_num	Searches by
1	Rows
2	Columns

Dir_num is the number 1 or 2 and specifies whether to search for the next or previous occurrence of text.

Dir_num	Searches for
1 or omitted	The next occurrence of text
2	The previous occurrence of text

Match_case is a logical value corresponding to the Match Case check box in the Find dialog box. If match_case is TRUE, Microsoft Excel matches characters exactly, including uppercase and lowercase; if FALSE or omitted, matching is not case-sensitive.

Remarks

- In Microsoft Excel for Windows, the dialog-box form of FORMULA.FIND is equivalent to pressing SHIFT+F5.
- If more than one cell is selected when you use FORMULA.FIND, Microsoft Excel searches only that selection.

FORMULA.FIND.NEXT, FORMULA.FIND.PREV

Finds the next and previous cells on the worksheet, as specified in the Find dialog box, and returns TRUE. (To see the Find dialog box, click Find on the Edit menu.) If a matching cell is not found, the functions return FALSE. For more information see FORMULA.FIND.

Syntax

FORMULA.FIND.NEXT( )

FORMULA.FIND.PREV( )

Related Functions

DATA.FIND   Selects records in a database that match the specified criteria

FORMULA.FIND   Finds text in a workbook

FORMULA.GOTO

Equivalent to clicking the Go To command on the Edit menu or to pressing F5. Scrolls through the worksheet and selects a named area or reference. Use FORMULA.GOTO to select a range on any open workbook; use SELECT to select a range on the active workbook.

Syntax

FORMULA.GOTO(reference, corner)

FORMULA.GOTO?(reference, corner)

Reference   specifies where to scroll and what to select.

- Reference should be either an external reference to a workbook, an R1C1-style reference in the form of text (see the second example following), or a name.
- If the Go To command has already been carried out, reference is optional. If reference is omitted, it is assumed to be the reference of the cells you selected before the previous Go To command or FORMULA.GOTO macro function was carried out. This feature distinguishes FORMULA.GOTO from SELECT.

Corner   is a logical value that specifies whether to scroll through the window so that the upper-left cell in reference is in the upper-left corner of the active window. If corner is TRUE, Microsoft Excel places reference in the upper-left corner of the window; if FALSE or omitted, Microsoft Excel scrolls through normally.

Tip   Microsoft Excel keeps a list of the cells you've selected with previous FORMULA.GOTO functions or Go To commands. When you use FORMULA.GOTO with GET.WORKSPACE(41), which returns a horizontal array of previous Go To selections, you can backtrack through multiple previous selections. See the last example below.

Remarks

- If you are recording a macro when you click the Go To command, the reference you enter in the Reference box of the Go To dialog box is recorded as text in the R1C1 reference style.

- If you are recording a macro when you double-click a cell that has precedents on another worksheet, Microsoft Excel records a FORMULA.GOTO function.

Examples

Each of the following macro formulas goes to cell A1 on the active worksheet:

```
FORMULA.GOTO(!$A$1)
```

```
FORMULA.GOTO("R1C1")
```

Each of the following macro formulas goes to the cells named Sales on the active worksheet and scrolls through the worksheet so that the upper-left corner of Sales is in the upper-left corner of the window:

```
FORMULA.GOTO(!Sales, TRUE)
```

```
FORMULA.GOTO("Sales", TRUE)
```

The following macro formula goes to the cells that were selected by the third most recent FORMULA.GOTO function or Go To command:

```
FORMULA.GOTO(INDEX(GET.WORKSPACE(41), 1, 3))
```

Related Functions

GOTO   Directs macro execution to another cell

HSCROLL   Horizontally scrolls through a sheet by percentage or by column or row number

SELECT   Selects a cell, worksheet object, or chart item

VSCROLL   Vertically scrolls through a sheet by percentage or by column or row number

FORMULA.REPLACE

Equivalent to clicking the Replace command on the Edit menu. Finds and replaces characters in cells on your worksheet.

Syntax

FORMULA.REPLACE(**find_text, replace_text**, look_at, look_by, active_cell, match_case)

FORMULA.REPLACE?(find_text, replace_text, look_at, look_by, active_cell, match_case)

Find_text   is the text you want to find. You can use the wildcard characters, question mark (?) and asterisk (*), in find_text. A question mark matches any single character; an asterisk matches any sequence of characters. If you want to find an actual question mark or asterisk, type a tilde (~) before the character.

Replace_text   is the text you want to replace find_text with.

Look_at is a number specifying whether you want find_text to match the entire contents of a cell or any string of matching characters.

Look_at	Looks for find_text
1 or omitted	As the entire contents of a cell
2	As part of the contents of a cell

Look_by is a number specifying whether to search horizontally (through rows) or vertically (through columns).

Look_by	Looks for find_text
1 or omitted	By rows
2	By columns

Active_cell is a logical value specifying the cells in which find_text is to be replaced.

- If active_cell is TRUE, find_text is replaced in the active cell only.
- If active_cell is FALSE, find_text is replaced in the entire selection, or, if the selection is a single cell, in the entire sheet.

Match_case is a logical value corresponding to the Match Case check box in the Replace dialog box. If match_case is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box. If match_case is omitted, the status of the check box is unchanged.

Remarks

- In FORMULA.REPLACE?, the dialog-box form of the function, omitted arguments are assumed to be the same arguments used in the previous replace operation. If there was no previous replace operation, omitted text arguments are assumed to be "" (empty text).
- The result of FORMULA.REPLACE must be a valid cell entry. For example, you cannot replace "=" with "= =" at the beginning of a formula.
- If more than a single cell is selected before you use FORMULA.REPLACE, only the selected cells are searched.

Related Function

FORMULA.FIND   Finds text in a workbook

FOURIER

Performs a Fourier transform.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

FOURIER(**inprng**, outrng, inverse, labels)

FOURIER?(inprng, outrng, inverse, labels)

Inprng   is the input range. The number of cells in the input range must be equal to a power of two (2, 4, 8, 16, ...).

Outrng   is the first cell in the output range or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Inverse   is a logical value. If TRUE, an inverse Fourier transform is performed. If FALSE or omitted, a forward Fourier transform is performed.

Labels   is a logical value.

- If labels is TRUE, then the first row or column of inprng contains labels.
- If labels is FALSE or omitted, all cells in inprng are considered data. Microsoft Excel generates appropriate data labels for the output table.

Related Function

SAMPLE   Samples data

FPOS

Sets the position of a file. The position of a file is where a character is read from or written to by an FREAD, FREADLN, FWRITE, or FWRITELN function. Use FPOS when you want to write characters to or read characters from specific locations. For example, to append text to the end of a file, you must set the position to the end of the file; otherwise, you might accidentally overwrite existing characters in the file.

Syntax

FPOS(**file_num**, position_num)

File_num   is the unique ID number of the file for which you want to set the position. File_num is returned by a previously executed FOPEN function. If file_num is not valid, FPOS returns the #VALUE! error value.

Position_num   is the location in the file that a character will be read from or written to.

- The first position in a file is 1, the location of the first byte.
- The last position in the file is the same as the value returned by FSIZE. For example, the last position in a file with 280 bytes is 280.

- If position_num is omitted, FPOS returns the current position of the file—that is, the number corresponding to where the next character will be read from or written to.

Whenever you read a character from or write a character to a file, the file's position is automatically incremented.

Examples

The following statement starts a loop that executes until the position in the open file identified as FileNumber reaches the end of the file:

```
=WHILE(FPOS(FileNumber)<=FSIZE(FileNumber))
```

Related Functions

FCLOSE	Closes a text file
FOPEN	Opens a file with the type of permission specified
FREAD	Reads characters from a text file
FREADLN	Reads a line from a text file
FWRITE	Writes characters to a text file
FWRITELN	Writes a line to a text file

FREAD

Reads characters from a file, starting at the current position in the file. (For more information about a file's position, see FPOS.) If FREAD is successful, it returns the text to the cell containing FREAD and set's the file's position to the start of the following line. If the end of the file is reached or if FREAD can't read the file, it returns the #N/A error value. Use FREAD instead of FREADLN when you need to read a specific number of characters from a text file.

Syntax

FREAD(**file_num, num_chars**)

File_num is the unique ID number of the file you want to read data from. File_num is returned by a previously executed FOPEN function. If file_num is not valid, FREAD returns the #VALUE! error value.

Num_chars specifies how many bytes to read from the file. FREAD can read up to 255 bytes at a time.

Example

The following function reads the next 200 bytes from the open file identified as FileNumber:

```
FREAD(FileNumber, 200)
```

Related Functions

FOPEN	Opens a file with the type of permission specified
FPOS	Sets the position in a text file

FREADLN Reads a line from a text file

FWRITE Writes characters to a text file

FREADLN

Reads characters from a file, starting at the current position in the file and continuing to the end of the line, placing the characters in the cell containing FREADLN. (For more information about a file's position, see FPOS.) If FREADLN is successful, it returns the text it read, up to but not including the carriage-return and linefeed characters at the end of the line (in Microsoft Excel for Windows) or the carriage-return character at the end of the line (in Microsoft Excel for the Macintosh). If the current file position is the end of the file or if FREADLN can't read the file, it returns the #N/A error value.

Syntax

FREADLN(**file_num**)

File_num is the unique ID number of the file you want to read data from. File_num is returned by a previously executed FOPEN function. If file_num is not valid, FREADLN returns the #VALUE! error value.

Example

The following function reads the next line from the open file identified as FileNumber:

```
FREADLN(FileNumber)
```

Related Functions

FOPEN Opens a file with the type of permission specified

FPOS Sets the position in a text file

FREAD Reads characters from a text file

FWRITE Writes characters to a text file

FWRITELN Writes a line to a text file

FREEZE.PANES

Equivalent to clicking the Freeze Panes or Unfreeze Panes command on the Window menu. Splits the active window into panes, creates frozen panes, or freezes or unfreezes existing panes. Use FREEZE.PANES to keep row or column titles on the screen while scrolling to other parts of the sheet.

Syntax

FREEZE.PANES(logical, col_split, row_split)

Logical is a logical value specifying which command FREEZE.PANES is equivalent to.

- If logical is TRUE, the function is equivalent to the Freeze Panes command. It freezes panes if they exist, or creates them, splits them at the specified position, and freezes them if they do not exist. If the panes are already frozen, FREEZE.PANES takes no action.

- If logical is FALSE, the function is equivalent to the Unfreeze Panes command. If no panes exist, FREEZE.PANES takes no action.
- If logical is omitted, FREEZE.PANES creates and then freezes panes if no panes exist, freezes existing panes if they're not currently frozen, or unfreezes existing panes if they're currently frozen.

Col_split   specifies where to split the window vertically and is measured in columns from the left of the window.

Row_split   specifies where to split the window horizontally and is measured in rows from the top of the window.

Col_split and row_split are ignored unless logical is TRUE and split panes do not exist.

Remarks

To create panes without freezing or unfreezing them, use the SPLIT function. You can freeze the panes later using the FREEZE.PANES function.

Related Functions

ACTIVATE	Switches to a window
SPLIT	Splits a window

FSIZE

Returns the number of bytes in a file. Use FSIZE to determine the size of the file, which is the same as the position of the last byte in the file.

Syntax

FSIZE(**file_num**)

File_num   is the unique ID number of the file whose size you want to know. File_num is returned by a previously executed FOPEN function. If file_num is not valid, FSIZE returns the #VALUE! error value.

Example

The following function returns the size in bytes of the open file identified as FileNumber:

```
FSIZE(FileNumber)
```

Related Functions

FOPEN	Opens a file with the type of permission specified
FPOS	Sets the position in a text file

FTESTV

Performs a two-sample F-test.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

FTESTV(**inprng1, inprng2**, outrng, labels)

FTESTV?(inprng1, inprng2. outrng, labels)

Inprng1 is the input range for the first data set.

Inprng2 is the input range for the second data set.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Labels is a logical value.

- If labels is TRUE, then the first row or column of inprng1 and inprng2 contain labels.
- If labels is FALSE or omitted, all cells in inprng1 and inprng2 are considered data. Microsoft Excel generates appropriate data labels for the output table.

FULL

Equivalent to pressing CTRL+F10 (full size) and CTRL+F5 (previous size) or double-clicking the title bar in Microsoft Excel for Windows version 3.0 or earlier. Equivalent to double-clicking the title bar or clicking the zoom box in Microsoft Excel for the Macintosh version 3.0 or earlier. This function is included only for macro compatibility. To perform the equivalent of a FULL(TRUE) function in Microsoft Excel version 4.0 or later, use the WINDOW.MAXIMIZE function. To perform the equivalent of a FULL(FALSE) function in Microsoft Excel version 4.0 or later, use the WINDOW.RESTORE function.

Syntax

FULL(logical)

FULL.SCREEN

Equivalent to clicking the Full Screen command on the View menu.

Syntax

FULL.SCREEN(logical)

Logical switches to full screen if TRUE or omitted; exits full screen mode if FALSE.

FUNCTION.WIZARD

Displays the Paste Function dialog box, which you can use to enter functions into cells.

Syntax

FUNCTION.WIZARD?( )

Remarks

If you know the function or formula that you want to insert into a cell, use the FORMULA function.

Related Function

FORMULA   Enters values into a cell or range or onto a chart

FWRITE

Writes text to a file, starting at the current position in that file. (For more information about a file's position, see FPOS.) If FWRITE can't write to the file, it returns the #N/A error value.

Syntax

FWRITE(**file_num, text**)

File_num   is the unique ID number of the file you want to write data to. File_num is returned by a previously executed FOPEN function. If file_num is not valid, FWRITE returns the #VALUE! error value.

Text   is the text you want to write to the file.

Example

The following function writes the current month to the open file identified as FileNumber:

```
FWRITE(FileNumber, TEXT(MONTH(NOW()),"mmmm"))
```

Related Functions

FOPEN   Opens a file with the type of permission specified

FPOS   Sets the position in a text file

FREAD   Reads characters from a text file

FWRITELN   Writes a line to a text file

FWRITELN

Writes text, followed by a carriage return and linefeed, to a file, starting at the current position in that file. (For more information about a file's position, see FPOS.) If FWRITELN can't write to the file, it returns the #N/A error value. Use FWRITELN instead of FWRITE when you want to append a carriage return and linefeed to each group of characters that you write to a text file.

Syntax

FWRITELN(**file_num, text**)

File_num is the unique ID number of the file you want to write data to. File_num is returned by a previously executed FOPEN function. If file_num is not valid, FWRITELN returns the #VALUE! error value.

Text is the text you want to write to the file.

Remarks

In Microsoft Excel for Windows, FWRITELN writes text followed by a carriage return and a line feed. In Microsoft Excel for the Macintosh, FWRITELN writes text followed by a carriage return only.

Example

The following function writes the current month to the open file identified as FileNumber and starts a new line in the file:

```
FWRITELN(FileNumber, TEXT(MONTH(NOW()),"mmmm"))
```

Related Functions

FOPEN	Opens a file with the type of permission specified
FPOS	Sets the position in a text file
FREAD	Reads characters from a text file
FWRITE	Writes characters to a text file

GALLERY.3D.AREA

Changes the format of the active chart to a 3-D area chart.

Syntax

GALLERY.3D.AREA(**type_num**)

GALLERY.3D.AREA?(type_num)

Type_num is the number of the 3-D Area format that you want to apply to the chart.

GALLERY.3D.BAR

Changes the active chart to a 3-D bar chart.

Syntax

GALLERY.3D.BAR(**type_num**)

GALLERY.3D.BAR?(type_num)

Type_num is the number of the 3-D Bar format that you want to apply to the chart.

GALLERY.3D.COLUMN

Changes the format of the active chart to a 3-D column chart.

Syntax

GALLERY.3D.COLUMN(**type_num**)

GALLERY.3D.COLUMN?(type_num)

Type_num is the number of the 3-D Column format that you want to apply to the chart.

GALLERY.3D.LINE

Changes the format of the active chart to a 3-D line chart.

Syntax

GALLERY.3D.LINE(**type_num**)

GALLERY.3D.LINE?(type_num)

Type_num is the number of the 3-D Line format that you want to apply to the chart.

GALLERY.3D.PIE

Changes the format of the active chart to a 3-D pie chart.

Syntax

GALLERY.3D.PIE(**type_num**)

GALLERY.3D.PIE?(type_num)

Type_num is the number of the 3-D Pie format that you want to apply to the chart.

GALLERY.3D.SURFACE

Changes the active chart to a 3-D surface chart.

Syntax

GALLERY.3D.SURFACE(**type_num**)

GALLERY.3D.SURFACE?(type_num)

Type_num is the number of the 3-D Surface format that you want to apply to the chart.

GALLERY.AREA

Changes the format of the active chart to an area chart.

Syntax

GALLERY.AREA(**type_num**, delete_overlay)

GALLERY.AREA?(type_num, delete_overlay)

Type_num is the number of a format in the AutoFormat dialog box when a chart is active dialog box that you want to apply to the area chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.BAR

Changes the format of the active chart to a bar chart.

Syntax

GALLERY.BAR(**type_num**, delete_overlay)

GALLERY.BAR?(type_num, delete_overlay)

Type_num is the number of the format that you want to apply to the bar chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.COLUMN

Changes the format of the active chart to a column chart.

Syntax

GALLERY.COLUMN(**type_num**, delete_overlay)

GALLERY.COLUMN?(type_num, delete_overlay)

Type_num is the number of the format you want to apply to the column chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.

- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.CUSTOM

Changes the format of the active chart to the custom format.

Syntax

GALLERY.CUSTOM(**name_text**)

Name_text   is the name of the custom template you want to apply.

Related Functions

ADD.CHART.AUTOFORMAT	Formats a chart using a custom gallery
DELETE.CHART.AUTOFORMAT	Deletes a custom gallery

GALLERY.DOUGHNUT

Changes the format of the active chart to a doughnut chart.

GALLERY.DOUGHNUT(**type_num**, delete_overlay)

GALLERY.DOUGHNUT?(type_num, delete_overlay)

Type_num   is the number of the format you want to apply to the doughnut chart.

Delete_overlay   is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.LINE

Changes the format of the active chart to a line chart.

Syntax

GALLERY.LINE(**type_num**, delete_overlay)

GALLERY.LINE?(type_num, delete_overlay)

Type_num   is the number of the format you want to apply to the line chart.

Delete_overlay   is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.PIE

Changes the format of the active chart to a pie chart.

Syntax

GALLERY.PIE(**type_num**, delete_overlay)

GALLERY.PIE?(type_num, delete_overlay)

Type_num is the number of the format you want to apply to the pie chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.RADAR

Changes the format of the active chart to a radar chart.

Syntax

GALLERY.RADAR(**type_num**, delete_overlay)

GALLERY.RADAR?(type_num, delete_overlay)

Type_num is the number of the format you want to apply to the radar chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.SCATTER

Changes the format of the active chart to an xy (scatter) chart.

Syntax

GALLERY.SCATTER(**type_num**, delete_overlay)

GALLERY.SCATTER?(type_num, delete_overlay)

Type_num is the number of the format you want to apply to the xy (scatter) chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GET.BAR

Returns the number of the active menu bar. There are two syntax forms of GET.BAR. Use syntax 1 to return information that you can use with other functions that manipulate menu bars. Use syntax 2 to return information that you can use with functions that add, delete, or alter menu commands.

Syntax 1 Returns the number of the active menu bar

Syntax 2 Returns the name or position number of a specified command on a menu or of a specified menu on a menu bar

GET.BAR SYNTAX 1

Returns the number of the active menu bar. There are two syntax forms of GET.BAR. Use syntax 1 to return information that you can use with other functions that manipulate menu bars. For a list of the ID numbers for Microsoft Excel's built-in menu bars, see ADD.COMMAND.

Syntax

GET.BAR( )

Example

The following macro formula assigns the name OldBar to the number of the active menu bar. This is useful if you will need to restore the current menu bar after displaying another custom menu bar.

```
SET.NAME("OldBar", GET.BAR())
```

Related Functions

ADD.BAR Adds a menu bar

SHOW.BAR   Displays a menu bar

GET.BAR Syntax 2   Returns the name or position number of a specified command on a menu or of a specified menu on a menu bar

GET.BAR SYNTAX 2

Returns the name or position number of a specified command on a menu or of a specified menu on a menu bar. There are two syntax forms of GET.BAR. Use syntax 2 to return information that you can use with functions that add, delete, or alter menu commands.

Syntax

GET.BAR(**bar_num, menu, command**, subcommand)

Bar_num   is the number of a menu bar containing the menu or command about which you want information. Bar_num can be the number of a built-in menu bar or the number returned by a previously run ADD.BAR function. For a list of the ID numbers for Microsoft Excel's built-in menu bars, see ADD.COMMAND.

Menu   is the menu on which the command resides or the menu whose name or position you want. Menu can be the name of the menu as text or the number of the menu. Menus are numbered starting with 1 from the left of the menu bar.

Command   is the command or submenu whose name or number you want returned. Command can be the name of the command from the menu as text, in which case the number is returned, or the number of the command from the menu, in which case the name is returned. Commands are numbered starting with 1 from the top of the menu. If command is 0, the name or position number of the menu is returned. If an ellipsis (...) follows a command name, such as the Open... command on the File menu, then you must include the ellipsis when referring to that command. See the following examples.

Subcommand   returns the name (if number is used for subcommand) or position (if name is used for subcommand) of a command on a submenu. If the command argument refers to an empty submenu, or is a command instead of a submenu, then using subcommand returns #N/A.

Remarks

- If an ampersand is used to indicate the access key in the name of a custom command, the ampersand is included in the name returned by GET.BAR. All built-in commands have an ampersand before the letter used as the access key.
- If the command name or position specified does not exist, GET.BAR returns the #N/A error value.

Examples

In the default worksheet and macro sheet menu bar:

`GET.BAR(10, "File", "Print...")` equals 14

`GET.BAR(10, "File", 14)` equals "&Print...^tCTRL+P" (where ^t is a tab character)

`GET.BAR(10, 1, "Open")` equals #N/A

`GET.BAR(10, 1, "Open...")` equals 2

Related Functions

ADD.COMMAND   Adds a command to a menu

DELETE.COMMAND   Deletes a command from a menu

GET.TOOLBAR   Retrieves information about a toolbar

RENAME.COMMAND   Changes the name of a command or menu

GETBAR Syntax 1   Returns the number of the active menu bar

GET.CELL

Returns information about the formatting, location, or contents of a cell. Use GET.CELL in a macro whose behavior is determined by the status of a particular cell.

Syntax

GET.CELL(**type_num**, reference)

Type_num   is a number that specifies what type of cell information you want. The following list shows the possible values of type_num and the corresponding results.

Type_num	**Returns**
1	Absolute reference of the upper-left cell in reference, as text in the current workspace reference style.
2	Row number of the top cell in reference.
3	Column number of the leftmost cell in reference.
4	Same as TYPE(reference).
5	Contents of reference.
6	Formula in reference, as text, in either A1 or R1C1 style depending on the workspace setting.
7	Number format of the cell, as text (for example, "m/d/yy" or "General").
8	Number indicating the cell's horizontal alignment: 1 = General 2 = Left

	3 = Center 4 = Right 5 = Fill 6 = Justify 7 = Center across cells
9	Number indicating the left-border style assigned to the cell: 0 = No border 1 = Thin line 2 = Medium line 3 = Dashed line 4 = Dotted line 5 = Thick line 6 = Double line 7 = Hairline
10	Number indicating the right-border style assigned to the cell. See type_num 9 for descriptions of the numbers returned.
11	Number indicating the top-border style assigned to the cell. See type_num 9 for descriptions of the numbers returned.
12	Number indicating the bottom-border style assigned to the cell. See type_num 9 for descriptions of the numbers returned.
13	Number from 0 to 18, indicating the pattern of the selected cell as displayed in the Patterns tab of the Format Cells dialog box, which appears when you click the Cells command on the Format menu. If no pattern is selected, returns 0.
14	If the cell is locked, returns TRUE; otherwise, returns FALSE.
15	If the cell's formula is hidden, returns TRUE; otherwise, returns FALSE.
16	A two-item horizontal array containing the width of the active cell and a logical value indicating whether the cell's width is set to change as the standard width changes (TRUE) or is a custom width (FALSE).
17	Row height of cell, in points.
18	Name of font, as text.

19 Size of font, in points.

20 If all the characters in the cell, or only the first character, are bold, returns TRUE; otherwise, returns FALSE.

21 If all the characters in the cell, or only the first character, are italic, returns TRUE; otherwise, returns FALSE.

22 If all the characters in the cell, or only the first character, are underlined, returns TRUE; otherwise, returns FALSE.

23 If all the characters in the cell, or only the first character, are struck through, returns TRUE; otherwise, returns FALSE.

24 Font color of the first character in the cell, as a number in the range 1 to 56. If font color is automatic, returns 0.

25 If all the characters in the cell, or only the first character, are outlined, returns TRUE; otherwise, returns FALSE. Outline font format is not supported by Microsoft Excel for Windows.

26 If all the characters in the cell, or only the first character, are shadowed, returns TRUE; otherwise, returns FALSE. Shadow font format is not supported by Microsoft Excel for Windows.

27 Number indicating whether a manual page break occurs at the cell:

0 = No break

1 = Row

2 = Column

3 = Both row and column

28 Row level (outline)

29 Column level (outline).

30 If the row containing the active cell is a summary row, returns TRUE; otherwise, returns FALSE.

31 If the column containing the active cell is a summary column, returns TRUE; otherwise, returns FALSE.

32 Name of the workbook and sheet containing the cell If the window contains only a single sheet that has the same name as the workbook without its extension, returns only the name of the book, in the form BOOK1.XLS. Otherwise, returns the name of the sheet in the form "[Book1]Sheet1".

33	If the cell is formatted to wrap, returns TRUE; otherwise, returns FALSE.
34	Left-border color as a number in the range 1 to 56. If color is automatic, returns 0.
35	Right-border color as a number in the range 1 to 56. If color is automatic, returns 0.
36	Top-border color as a number in the range 1 to 56. If color is automatic, returns 0.
37	Bottom-border color as a number in the range 1 to 56. If color is automatic, returns 0.
38	Shade foreground color as a number in the range 1 to 56. If color is automatic, returns 0.
39	Shade background color as a number in the range 1 to 56. If color is automatic, returns 0.
40	Style of the cell, as text.
41	Returns the formula in the active cell without translating it (useful for international macro sheets).
42	The horizontal distance, measured in points, from the left edge of the active window to the left edge of the cell. May be a negative number if the window is scrolled beyond the cell.
43	The vertical distance, measured in points, from the top edge of the active window to the top edge of the cell. May be a negative number if the window is scrolled beyond the cell.
44	The horizontal distance, measured in points, from the left edge of the active window to the right edge of the cell. May be a negative number if the window is scrolled beyond the cell.
45	The vertical distance, measured in points, from the top edge of the active window to the bottom edge of the cell. May be a negative number if the window is scrolled beyond the cell.
46	If the cell contains a text note, returns TRUE; otherwise, returns FALSE.
47	If the cell contains a sound note, returns TRUE; otherwise, returns FALSE.
48	If the cells contains a formula, returns TRUE; if a constant, returns FALSE.

49	If the cell is part of an array, returns TRUE; otherwise, returns FALSE.
50	Number indicating the cell's vertical alignment: 1 = Top 2 = Center 3 = Bottom 4 = Justified
51	Number indicating the cell's vertical orientation: 0 = Horizontal 1 = Vertical 2 = Upward 3 = Downward
52	The cell prefix (or text alignment) character, or empty text ("") if the cell does not contain one.
53	Contents of the cell as it is currently displayed, as text, including any additional numbers or symbols resulting from the cell's formatting.
54	Returns the name of the PivotTable report containing the active cell.
55	Returns the position of a cell within the PivotTable report. 0 = Row header 1 = Column header 2 = Page header 3 = Data header 4 = Row item 5 = Column item 6 = Page item 7 = Data item 8 = Table body
56	Returns the name of the field containing the active cell reference if inside a PivotTable report.
57	Returns TRUE if all the characters in the cell, or only the first character, are formatted with a superscript font; otherwise, returns FALSE.

58	Returns the font style as text of all the characters in the cell, or only the first character as displayed in the Font tab of the Format Cells dialog box: for example, "Bold Italic".
7	Returns the number for the underline style: 1 = None 2 = Single 3 = Double 4 = Single accounting 5 = Double accounting
60	Returns TRUE if all the characters in the cell, or only the first character, are formatted with a subscript font; otherwise, it returns FALSE.
61	Returns the name of the PivotTable item for the active cell, as text.
62	Returns the name of the workbook and the current sheet in the form "[Book1]Sheet1".
63	Returns the fill (background) color of the cell.
64	Returns the pattern (foreground) color of the cell.
65	Returns TRUE if the Add Indent alignment option is on (Far East versions of Microsoft Excel only); otherwise, it returns FALSE.
66	Returns the book name of the workbook containing the cell in the form BOOK1.XLS.

Reference is a cell or a range of cells from which you want information.

- If reference is a range of cells, the cell in the upper-left corner of the first range in reference is used.
- If reference is omitted, the active cell is assumed.

Tip Use GET.CELL(17) to determine the height of a cell and GET.CELL(44) - GET.CELL(42) to determine the width.

Examples

The following macro formula returns TRUE if cell B4 on sheet Sheet1 is bold:

```
GET.CELL(20, Sheet1!$B$4)
```

You can use the information returned by GET.CELL to initiate an action. The following macro formula runs a custom function named BoldCell if the GET.CELL formula returns FALSE:

```
IF(GET.CELL(20, Sheet1!$B$4), , BoldCell())
```

Related Functions

ABSREF   Returns the absolute reference of a range of cells to another range

ACTIVE.CELL   Returns the reference of the active cell

GET.FORMULA   Returns the contents of a cell

GET.NAME   Returns the definition of a name

GET.NOTE   Returns characters from a note

RELREF   Returns a relative reference

GET.CHART.ITEM

Returns the vertical or horizontal position of a point on a chart item. Use these position numbers with FORMAT.MOVE and FORMAT.SIZE to change the position and size of chart items. Position is measured in points; a point is 1/72nd of an inch.

Syntax

GET.CHART.ITEM(**x_y_index**, point_index, item_text)

X_y_index   is a number specifying which of the coordinates you want returned.

X_y_index	Coordinate returned
1	Horizontal coordinate
2	Vertical coordinate

Point_index   is a number specifying the point on the chart item. These indexes are described below. If point_index is omitted, it is assumed to be 1.

- If the specified item is a point, point_index must be 1.
- If the specified item is any line other than a data line, use the following values for point_index.

Point_index	Chart item position
1	Lower or left

2	Upper or right

- If the selected item is a legend, plot area, chart area, or an area in an area chart, use the following values for point_index.

Point_index	**Chart item position**
1	Upper left
2	Upper middle
3	Upper right
4	Right middle
5	Lower right
6	Lower middle
7	Lower left
8	Left middle

- If the selected item is an arrow in Microsoft Excel 4.0, use the following values for point_index. In Microsoft Excel version 5.0 or later, arrows are named lines, and the arrowhead position returned is equivalent to the end of a line where the arrowhead begins.

Point_index	**Chart item position**
1	Arrow shaft
2	Arrowhead

- If the selected item is a pie slice, use the following values for point_index.

Point_index	Chart item position
1	Outermost counterclockwise point
2	Outer center point
3	Outermost clockwise point
4	Midpoint of the most clockwise radius
5	Center point
6	Midpoint of the most counterclockwise radius

Item_text is a selection code that specifies which item of a chart to select. See the chart form of SELECT for the item_text codes to use for each item of a chart.

- If item_text is omitted, it is assumed to be the currently selected item.
- If item_text is omitted and no item is selected, GET.CHART.ITEM returns the #VALUE! error value.

Remarks

If the specified item does not exist, or if a chart is not active when the function is carried out, the #VALUE! error value is returned.

Examples

The following macro formulas return the horizontal and vertical locations, respectively, of the top of the main-chart value axis:

```
GET.CHART.ITEM(1, 2, "Axis 1")

GET.CHART.ITEM(2, 2, "Axis 1")
```

You could then use FORMAT.MOVE to move a floating text item to the position returned by these two formulas.

Related Functions

GET.DOCUMENT Returns information about a workbook

GET.FORMULA Returns the contents of a cell

GET.DEF

Returns the name, as text, that is defined for a particular area, value, or formula in a workbook. Use GET.DEF to get the name corresponding to a definition. To get the definition of a name, use GET.NAME.

Syntax

GET.DEF(**def_text**, document_text, type_num)

Def_text can be anything you can define a name to refer to, including a reference, a value, an object, or a formula.

- References must be given in R1C1 style, such as "R3C5".
- If def_text is a value or formula, it is not necessary to include the equal sign that is displayed in the Refers To box in the Define Name dialog box, which appears when you choose the Name command from the Define submenu on the Insert Menu.
- If there is more than one name for def_text, GET.DEF returns the first name. If no name matches def_text, GET.DEF returns the #NAME? error value.

Document_text specifies the sheet or macro sheet that def_text is on. If document_text is omitted, it is assumed to be the active macro sheet.

Type_num is a number from 1 to 3 specifying which types of names are returned.

Type_num	Returns
1 or omitted	Normal names only
2	Hidden names only
3	All names

Examples

If the specified range in Sheet4 is named Sales, the following macro formula returns "Sales":

```
GET.DEF("R2C2:R9C6", "Sheet4")
```

If the value 100 in Sheet4 is defined as Constant, the following macro formula returns "Constant":

```
GET.DEF("100", "Sheet4")
```

If the specified formula in Sheet4 is named SumTotal, the following macro formula returns "SumTotal":

```
GET.DEF("SUM(R1C1:R10C1)", "Sheet4")
```

If 3 is defined as the hidden name Counter on the active macro sheet, the following macro formula returns "Counter":

```
GET.DEF("3", , 2)
```

Related Functions

GET.CELL   Returns information about the specified cell

GET.NAME   Returns the definition of a name

GET.NOTE   Returns characters from a note

NAMES   Returns the names defined on a workbook

GET.DOCUMENT

Returns information about a sheet in a workbook.

Syntax

GET.DOCUMENT(**type_num**, name_text)

Type_num   is a number that specifies what type of information you want. The following lists show the possible values of type_num and the corresponding results.

Type_num	Returns
1	Returns the name of the workbook and worksheet as text. If there is only one sheet in the workbook and the sheet name is the same as the workbook name less any extension, returns the name of the book. The book name does not include the drive, directory or folder, or window number. Otherwise, returns the book and sheet name in the form "[BOOK1.XLS]Sheet1". It is usually best to use GET.DOCUMENT(76) and GET.DOCUMENT(88) to return the name of the active worksheet and the active workbook.
2	Path of the directory or folder containing name_text, as text. If the workbook name_text hasn't been saved yet, returns the #N/A error value.
3	Number indicating the type of sheet. If name_text is a sheet, then the return value is one of the following numbers. If name_text is a book, then the return value is always 5. If name_text is omitted, then the sheet type is

	returned. If the book has one sheet that is named the same as the book, then the sheet type is returned. 1 = Worksheet 2 = Chart 3 = Macro sheet 4 = Info window if active 5 = Reserved 6 = Module 7 = Dialog
4	If changes have been made to the sheet since it was last saved, returns TRUE; otherwise, returns FALSE.
5	If the sheet is read-only, returns TRUE; otherwise, returns FALSE.
6	If the sheet is password protected, returns TRUE; otherwise, returns FALSE.
7	If cells in a sheet, the contents of a sheet, or the series in a chart are protected, returns TRUE; otherwise, returns FALSE.
8	If the workbook windows are protected, returns TRUE; otherwise, returns FALSE.

The next four values of type_num apply only to charts.

Type_num	**Returns**
9	Number indicating the type of the main chart: 1 = Area 2 = Bar 3 = Column 4 = Line 5 = Pie 6 = XY (scatter) 7 = 3-D area 8 = 3-D column

	9 = 3-D line 10 = 3-D pie 11 = Radar 12 = 3-D bar 13 = 3-D surface 14 = Doughnut
10	Number indicating the type of the overlay chart. Same as 1, 2, 3, 4, 5, 6, 11, and 14 for main chart above. If there is no overlay chart, returns the #N/A error value.
11	Number of series in the main chart.
12	Number of series in the overlay chart.

The next values of type_num apply to worksheets and macro sheets and to charts when appropriate.

Type_num	**Returns**
9	Number of the first used row. If the sheet is empty, returns 0.
10	Number of the last used row. If the sheet is empty, returns 0.
11	Number of the first used column. If the sheet is empty, returns 0.
12	Number of the last used column. If the sheet is empty, returns 0.
13	Number of windows.
14	Number indicating calculation mode: 1 = Automatic 2 = Automatic except tables 3 = Manual

15	If the Iteration check box is selected in the Calculation tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
16	Maximum number of iterations.
17	Maximum change between iterations.
18	If the Update Remote References check box is selected in the Calculation tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
19	If the Precision As Displayed check box is selected in the Calculation tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
20	If the 1904 Date System check box is selected in the Calculation tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.

Type_num values of 21 through 29 correspond to the four default fonts in previous versions of Microsoft Excel. These values are provided only for macro compatibility.

The next values of type_num apply to worksheets and macro sheets, and to charts if indicated.

Type_num	**Returns**
30	Horizontal array of consolidation references for the current sheet, in the form of text. If the list is empty, returns the #N/A error value.
31	Number from 1 to 11, indicating the function used in the current consolidation. The function that corresponds to each number is listed under the CONSOLIDATE function. The default function is SUM.
32	Three-item horizontal array indicating the status of the check boxes in the Data Consolidate dialog box. An item is TRUE if the check box is selected or FALSE if the check box is cleared. The first item indicates the Top Row check box, the second the Left Column check box, and the third the Create Links To Source Data check box.

33	If the Recalculate Before Save check box is selected in the Calculation tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
34	If the workbook is read-only recommended, returns TRUE; otherwise, returns FALSE.
35	If the workbook is write-reserved, returns TRUE; otherwise, returns FALSE.
36	If the workbook has a write-reservation password and it is opened with read/write permission, returns the name of the user who originally saved the file with the write-reservation password. If the file is opened as read-only, or if a password has not been added to the workbook, returns the name of the current user.
37	Number corresponding to the file type of the workbook as displayed in the Save As dialog box. See the SAVE.AS function for a list of all the file types that Microsoft Excel recognizes.
38	If the Summary Rows Below Detail check box is selected in the Outline dialog box, returns TRUE; otherwise, returns FALSE.
39	If the Summary Columns To Right Of Detail check box is selected in the Outline dialog box, returns TRUE; otherwise, returns FALSE.
40	If the Always Create Backup check box is selected in the Save Options dialog box, returns TRUE; otherwise, returns FALSE.
41	Number from 1 to 3 indicating whether objects are displayed: 1 = All objects are displayed 2 = Placeholders for pictures and charts 3 = All objects are hidden

42	Horizontal array of all objects in the sheet. If there are no objects, returns the #N/A error value.
43	If the Save External Link Values check box is selected in the Calculation tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
44	If objects in a workbook are protected, returns TRUE; otherwise, returns FALSE.
45	A number from 0 to 3 indicating how windows are synchronized: 0 = Not synchronized 1 = Synchronized horizontally 2 = Synchronized vertically 3 = Synchronized horizontally and vertically
46	A seven-item horizontal array of print settings that can be set by the LINE.PRINT macro function: Setup text Left margin Right margin Top margin Bottom margin Page length A logical value indicating whether output will be formatted (TRUE) or unformatted (FALSE) when printed
47	If the Transition Formula Evaluation check box is selected in the Transition tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
48	The standard column width setting.

The next values of type_num correspond to printing and page settings.

Type_num	**Returns**
49	The starting page number, or the #N/A error value if none is specified or if "Auto" is entered in the First page Number text box on the Page tab of the Page Setup dialog box.

50	The total number of pages that would be printed based on current settings, excluding notes, or 1 if the document is a chart.
51	The total number of pages that would be printed if you print only notes, or the #N/A error value if the document is a chart.
52	Four-item horizontal array indicating the margin settings (left, right, top, bottom) in the currently specified units.
53	A number indicating the orientation: 1 = Portrait 2 = Landscape
54	The header as a text string, including formatting codes.
55	The footer as a text string, including formatting codes.
56	Horizontal array of two logical values corresponding to horizontal and vertical centering.
57	If row or column headings are to be printed, returns TRUE; otherwise, returns FALSE.
58	If gridlines are to be printed, returns TRUE; otherwise, returns FALSE.
59	If the sheet is printed in black and white only, returns TRUE; otherwise, returns FALSE.
60	A number from 1 to 3 indicating how the chart will be sized when it's printed: 1 = Size on screen 2 = Scale to fit page 3 = Use full page
61	A number indicating the pagination order: 1 = Down, then over 2 = Over, then down Returns the #N/A error value if the document is a chart.

62	Percentage of reduction or enlargement, or 100% if none is specified. Returns the #N/A error value if not supported by the current printer or if the document is a chart.
63	A two-item horizontal array indicating the number of pages to which the printout should be scaled to fit, with the first item equal to the width (or #N/A if no width scaling is specified) and the second item equal to the height (or #N/A if no height scaling is specified). #N/A is also returned if the document is a chart.
64	An array of row numbers corresponding to rows that are immediately below a manual or automatic page break.
65	An array of column numbers corresponding to columns that are immediately to the right of a manual or automatic page break.

Note GET.DOCUMENT(62) and GET.DOCUMENT(63) are mutually exclusive. If one returns a value, then the other returns the #N/A error value.

The next values of type_num correspond to various workbook settings.

Type_num	**Returns**
66	In Microsoft Excel for Windows, if the Transition Formula Entry check box is selected in the Transition tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
67	Microsoft Excel version 5.0 or later always returns TRUE here.
68	Microsoft Excel version 5.0 or later always returns the book name.
69	Returns TRUE if Page Breaks is chosen in the View tab of the Options dialog box; otherwise, returns FALSE.
70	Returns the names of all PivotTable reports in the current sheet as a horizontal array.
71	Returns an horizontal array of all the styles in a workbook.

72	Returns an horizontal array of all chart types displayed on the current sheet.
73	Returns an array of the number of series in each chart of the current sheet.
74	Returns the object ID of the control that currently has the focus on a running user-defined dialog (based on the dialog sheet).
75	Returns the object ID of the object that is the current default button on a running user-defined dialog (based on the dialog sheet).
76	Returns the name of the active sheet or macro sheet in the form [Book1]Sheet1.
77	In Microsoft Excel for Windows, returns the paper size, as integer: 1 = Letter 8.5 x 11 in 2 = Letter Small 8.5 x 11 in 5 = Legal 8.5 x 14 in 9 = A4 210 x 297 mm 10 = A4 Small 210 x 297 mm 13 = B5 182 x 257 mm 18 = Note 8.5 x 11 in
78	Returns the print resolution, as a horizontal array of two numbers.
79	Returns TRUE if the Draft Quality check box has been selected from the sheet tab in the Page Setup dialog box; otherwise, returns FALSE.
80	Returns TRUE if the Comments checkbox has been selected on the Sheet tab in the Page Setup dialog box; otherwise, returns FALSE.
81	Returns the Print Area from the Sheet tab of the Page Setup dialog box as a cell reference.

82	Returns the Print Titles from the Sheet tab of the Page Setup dialog box as an array of cell references.
83	Returns TRUE if the worksheet is protected for scenarios; otherwise, returns FALSE.
84	Returns the value of the first circular reference on the sheet, or #N/A if there are no circular references.
85	Returns the advanced filter mode state of the sheet. This is the mode without drop-down arrows on top. Returns TRUE if the list has been filtered by clicking Filter, then Advanced Filter on the Data menu. Otherwise, returns FALSE.
86	Returns the automatic filter mode state of the sheet. This is the mode with drop-down arrows on top. Returns TRUE if you have chosen Filter, then AutoFilter from the Data menu and the filter drop-down arrows are displayed. Otherwise, returns FALSE.
87	Returns the position number of the sheet. The first sheet is position 1. Hidden sheet are included in the count.
88	Returns the name of the active workbook in the form "Book1".

Name_text is the name of an open workbook. If name_text is omitted, it is assumed to be the active workbook.

Examples

The following macro formula returns TRUE if the contents of the active workbook are protected:

```
GET.DOCUMENT(7)
```

In Microsoft Excel for Windows, the following macro formula returns the number of windows in SALES.XLS:

```
GET.DOCUMENT(13, "SALES.XLS")
```

In Microsoft Excel for the Macintosh, the following macro formula returns 3 if the overlay chart on SALES CHART is a column chart:

```
GET.DOCUMENT(10, "SALES CHART")
```

To find out if SHEET1 is password-protected and if its contents and windows are protected, enter the following formula in a three-cell horizontal array:

```
GET.DOCUMENT({6, 7, 8}, "SHEET1")
```

Related Functions

GET.CELL   Returns information about the specified cell

GET.WINDOW   Returns information about a window

GET.WORKSPACE   Returns information about the workspace

GET.FORMULA

Returns the contents of a cell as they would appear in the formula bar. The contents are given as text, for example, "=2*PI()/360". If the formula contains references, they are returned as R1C1-style references, such as "=RC[1]*(1+R1C1)". Use GET.FORMULA to get a formula from a cell in order to edit its arguments. Use GET.CELL(6) to get a formula in either A1 or R1C1 format, depending on the workspace setting.

Syntax

GET.FORMULA(**reference**)

Reference   is a cell or range of cells on a sheet or macro sheet.

- If a range of cells is selected, GET.FORMULA returns the contents of the upper-left cell in reference.
- Reference can be an external reference.
- Reference can be the object identifier of a picture created by the camera tool.
- Reference can also be a reference to a chart series in the form "Sn" where n is the number of the series. When a chart series is specified, GET.FORMULA returns the series formula using R1C1-style references.

Tip   If you want to get the formula in the active cell, use the ACTIVE.CELL function as the reference argument.

Examples

If cell A3 on the active sheet contains the number 523, then:

`GET.FORMULA(!$A$3)` equals "523"

If cell C2 on the active sheet contains the formula =B2*(1+$A$1), then:

`GET.FORMULA(!$C$2)` equals "=RC[-1]*(1+R1C1)"

The following macro formula returns the contents of the active cell on the active sheet:

```
GET.FORMULA(ACTIVE.CELL())
```

Related Functions

GET.CELL   Returns information about the specified cell

GET.DEF   Returns a name matching a definition

GET.NAME   Returns the definition of a name

GET.NOTE   Returns characters from a comment

GET.LINK.INFO

Returns information about the specified link. Use GET.LINK.INFO to get information about the update settings of a link.

Syntax

GET.LINK.INFO(**link_text, type_num**, type_of_link, reference)

Link_text   is the path of the link as displayed in the Links dialog box, which appears when you choose the Links command from the Edit menu. The path to the file you wish to return DDE information on must be surrounded by single quotes.

Type_num   is a number that specifies what type of information about the currently selected link to return. Type_num 2 applies only to publishers and subscribers in Microsoft Excel for the Macintosh.

Type_num	**Returns**
1	If the link is set to automatic update, returns 1; otherwise 2.
2	Date of the latest edition as a serial number. Returns #N/A if link_text is not a publisher or a subscriber.

Type_of_link   is a number from 1 to 6 that specifies what type of link you want to get information about.

Type_of_link	**Link document type**
1	Not applicable
2	DDE link (Microsoft Windows)
3	Not applicable
4	Not applicable
5	Publisher (Macintosh)
6	Subscriber (Macintosh)

Reference   specifies the cell range in R1C1 format of the publisher or subscriber that you want information about. Reference is required if you have more than one publisher or subscriber of a single edition name on the active workbook. Use reference to specify the location of the subscriber you want to return information about. If the subscriber is a picture, or if the publisher is an embedded chart, reference is the number of the object as displayed in the Name box.

Remarks

- If Microsoft Excel cannot find link_text, or if type_of_link does not match the link specified by link_text, GET.LINK.INFO returns the #VALUE! error value.
- If you have more than one subscriber to the edition link_text or if the same area is published more than once, you must specify reference.

Example

In Microsoft Excel for Windows, the following macro formula returns information about a DDE link to a Microsoft Word for Windows document. The document is named NEWPROD.DOC.

```
GET.LINK.INFO("WinWord|'C:\WINWORD\NEWPROD.DOC'!DDE_LINK1", 1, 2)
```

In Microsoft Excel for the Macintosh, the following macro formula returns information about a link to a publisher defined in cells A1:C3 on a workbook named New Products.

```
GET.LINK.INFO("A1:C3 New Products Edition #1", 2, 5, "'New
Products'!R1C1:R3C3")
```

Related Functions

CREATE.PUBLISHER   Creates a publisher from the selection

SUBSCRIBE.TO   Inserts contents of an edition into the active workbook

UPDATE.LINK   Updates a link to another workbook

GET.NAME

Returns the definition of a name as it appears in the Refers To box of the Define Name dialog box, which appears when you choose the Define command from the Name submenu on the Insert menu. If the definition contains references, they are given as R1C1-style references. Use GET.NAME to check the value defined by a name. To get the name corresponding to a definition, use GET.DEF.

Syntax

GET.NAME(**name_text**, info_type)

Name_text   can be a name defined on the macro sheet; an external reference to a name defined on the active workbook, for example, "!Sales"; or an external reference to a name defined on a particular open workbook, for example, "[Book1]SHEET1!Sales". Name_text can also be a hidden name.

Info_type   specifies the type of information to return about the name. If 1 or omitted, the definition is returned. If 2, returns TRUE if the name is defined for just the sheet, FALSE if the name is defined for the entire workbook.

Remarks

If the Contents check box has been selected in the Protect Sheet dialog box to protect the workbook containing the name, GET.NAME returns the #N/A error value. To see the Protect Sheet dialog box, choose the Protect Sheet command on the Protection submenu from the Tools menu.

Examples

If the name Sales on a macro sheet is defined as the number 523, then:

`GET.NAME("Sales")` equals "=523"

If the name Profit on the active sheet is defined as the formula =Sales-Costs, then:

`GET.NAME("!Profit")` equals "=Sales-Costs"

If the name Database on the active sheet is defined as the range A1:F500, then:

`GET.NAME("!Database")` equals "=R1C1:R500C6"

Related Functions

DEFINE.NAME   Defines a name on the active or macro sheet

GET.CELL   Returns information about the specified cell

GET.DEF   Returns a name matching a definition

NAMES   Returns the names defined in a workbook

SET.NAME   Defines a name as a value

GET.NOTE

Returns characters from a comment.

Syntax

GET.NOTE(cell_ref, start_char, num_chars)

Cell_ref   is the cell to which the note is attached. If cell_ref is omitted, the comment attached to the active cell is returned.

Start_char   is the number of the first character in the comment to return. If start_char is omitted, it is assumed to be 1, the first character in the comment.

Num_chars   is the number of characters to return. Num_chars must be less than or equal to 255. If num_chars is omitted, it is assumed to be the length of the comment attached to cell_ref.

Examples

The following macro formula returns the first 200 characters in the comment attached to cell A3 on the active sheet:

```
GET.NOTE(!$A$3, 1, 200)
```

In Microsoft Excel for Windows, the following macro formula returns the 10th through the 39th characters of the comment attached to cell C2 on SALES.XLS:

```
GET.NOTE("[SALES.XLS]Sheet1!R2C3", 10, 30)
```

In Microsoft Excel for the Macintosh, the following macro formula returns the 10th through the 39th characters of the comment attached to cell C2 on SALES:

```
GET.NOTE("[SALES]Sheet1!R2C3", 10, 30)
```

Use GET.NOTE with the NOTE function to move the contents of a comment to a cell or text box or to another comment attached to a cell:

```
NOTE(GET.NOTE(!$B$10),ACTIVE.CELL())
```

Related Functions

GET.CELL   Returns information about the specified cell

NOTE   Creates or changes a comment.

GET.OBJECT

Returns information about the specified object. Use GET.OBJECT to return information you can use in other macro formulas that manipulate objects.

Syntax

GET.OBJECT(**type_num**, object_id_text, start_num, count_num, item_index)

Type_num   is a number specifying the type of information you want returned about an object. GET.OBJECT returns the #VALUE! error value (and the macro is halted) if an object isn't specified or if more than one object is selected.

Type_num	**Returns**
2	If the object is locked, returns TRUE; otherwise FALSE.
3	Z-order position (layering) of the object; that is, the relative position of the overlapping objects, starting with 1 for the object that is most under the others.
4	Reference of the cell under the upper-left corner of the object as text in R1C1 reference style; for a line or arc, returns the start point.
5	X offset from the upper-left corner of the cell under the upper-left corner of the object, measured in points.
6	Y offset from the upper-left corner of the cell under the upper-left corner of the object, measured in points.

7	Reference of the cell under the lower-right corner of the object as text in R1C1 reference style; for a line or arc, returns the end point.
8	X offset from the upper-left corner of the cell under the lower-right corner of the object, measured in points.
9	Y offset from the upper-left corner of the cell under the lower-right corner of the object, measured in points.
10	Name, including the filename, of the macro assigned to the object. If no macro is assigned, returns FALSE.
11	Number indicating how the object moves and sizes: 1 = Object moves and sizes with cells 2 = Object moves with cells 3 = Object is fixed

Values 12 to 21 for type_num apply only to text boxes and buttons. If another type of object is selected, GET.OBJECT returns the #VALUE! error value.

Type_num	Returns
12	Text starting at start_num for count_num characters.
13	Font name of all text starting at start_num for count_num characters. If the text contains more than one font name, returns the #N/A error value.
14	Font size of all text starting at start_num for count_num characters. If the text contains more than one font size, returns the #N/A error value.
15	If all text starting at start_num for count_num characters is bold, returns TRUE. If text contains only partial bold formatting, returns the #N/A error value.
16	If all text starting at start_num for count_num characters is italic, returns TRUE. If text contains only partial italic formatting, returns the #N/A error value.

17	If all text starting at start_num for count_num characters is underlined, returns TRUE. If text contains only partial underline formatting, returns the #N/A error value.
18	If all text starting at start_num for count_num characters is struck through, returns TRUE. If text contains only partial struck-through formatting, returns the #N/A error value.
19	In Microsoft Excel for the Macintosh, if all text starting at start_num for count_num characters is outlined, returns TRUE. If text contains only partial outline formatting, returns the #N/A error value. Always returns FALSE in Microsoft Excel for Windows.
20	In Microsoft Excel for the Macintosh, if all text starting at start_num for count_num characters is shadowed, returns TRUE. If text contains only partial shadow formatting, returns the #N/A error value. Always returns FALSE in Microsoft Excel for Windows.
21	Number from 0 to 56 indicating the color of all text starting at start_num for count_num characters; if color is automatic, returns 0. If more than one color is used, returns the #N/A error value.

Values 22 to 25 for type_num also apply only to text boxes and buttons. If another type of object is selected, GET.OBJECT returns the #N/A error value.

Type_num	**Returns**
23	Number indicating the vertical alignment of text: 1 = Top 2 = Center 3 = Bottom 4 = Justified
24	Number indicating the orientation of text: 0 = Horizontal 1 = Vertical 2 = Upward 3 = Downward
25	If button or text box is set to automatic sizing, returns TRUE; otherwise FALSE.

The following values for type_num apply to all objects, except where indicated.

Type_num	Returns
27	Number indicating the type of the border or line: 0 = Custom 1 = Automatic 2 = None
28	Number indicating the style of the border or line as shown in the Patterns tab in the Format Objects dialog box: 0 = None 1 = Solid line 2 = Dashed line 3 = Dotted line 4 = Dashed dotted line 5 = Dashed double-dotted line 6 = 50% gray line 7 = 75% gray line 8 = 25% gray line
29	Number from 0 to 56 indicating the color of the border or line; if the border is automatic, returns 0.
30	Number indicating the weight of the border or line: 1 = Hairline 2 = Thin 3 = Medium 4 = Thick
31	Number indicating the type of fill: 0 = Custom 1 = Automatic 2 = None
32	Number from 1 to 18 indicating the fill pattern as shown in the Format Object dialog box.
33	Number from 0 to 56 indicating the foreground color of the fill pattern; if the fill is automatic, returns 0. If the object is a line, returns the #N/A error value.

34 Number from 0 to 56 indicating the background color of the fill pattern; if the fill is automatic, returns 0. If the object is a line, returns the #N/A error value.

35 Number indicating the width of the arrowhead:
1 = Narrow
2 = Medium
3 = Wide
If the object is not a line, returns the #N/A error value.

36 Number indicating the length of the arrowhead:
1 = Short
2 = Medium
3 = Long
If the object is not a line, returns the #N/A error value.

37 Number indicating the style of the arrowhead:
1 = No head
2 = Open head
3 = Closed head
4 = Open double-ended head
5 = Closed double-ended head
If the object is not a line, returns the #N/A error value.

38 If the border has round corners, returns TRUE; if the corners are square, returns FALSE. If the object is a line, returns the #N/A error value.

39 If the border has a shadow, returns TRUE; if the border has no shadow, returns FALSE. If the object is a line, returns the #N/A error value.

40 If the Lock Text check box in the Protection Tab of the Format Object dialog box is selected, returns TRUE; otherwise FALSE.

41 If objects are set to be printed, returns TRUE; otherwise FALSE.

42 The horizontal distance, measured in points, from the left edge of the active window to the left edge of the object. May be a negative number if the window is scrolled beyond the object.

43 The vertical distance, measured in points, from the top edge of the active window to the top edge of the object. May be a negative number if the window is scrolled beyond the object.

44 The horizontal distance, measured in points, from the left edge of the active window to the right edge of the object. May be a negative number if the window is scrolled beyond the object.

45 The vertical distance, measured in points, from the top edge of the active window to the bottom edge of the object. May be a negative number if the window is scrolled beyond the object.

46 The number of vertices in a polygon, or the #N/A error value if the object is not a polygon.

47 A count_num by 2 array of vertex coordinates starting at start_num in a polygon's array of vertices.

48 If the object is a text box, returns the cell reference that the text box is linked to. If the object is a control on a worksheet, returns the cell reference that the control's value is linked to. This information is returned as a string.

49 Returns the ID number of the object. For example, "Rectangle 5" returns 5. Note that the name of the object may not have this index in it if the object has been renamed by the user.

50 Returns the object's classname. For example, "Rectangle".

51 Returns the object name. By default, object names are the classname followed by the ID. For example, "Rectangle 1" is an object name, of which "Rectangle" is the classname, and 1 is the ID number. The object can also be renamed, in which case the name picked by the user is returned.

52 Returns the distance from cell A1 to the Left of the object bounding rectangle in points

53 Returns the distance from Cell A1 to the top of the object bounding rectangle in points

54 Returns the width of object bounding rectangle in points

55 Returns the height of object bounding rectangle in points

56 If the object is enabled, returns TRUE; otherwise, it returns FALSE.

57 Returns the shortcut key assignment for the control object, as text.

58 Returns TRUE is the button control on a dialog sheet is the default button of the dialog; otherwise, returns FALSE

59 Returns TRUE if the button control on the dialog sheet is clicked when the user presses the ESCAPE Key; otherwise, returns FALSE.

60 Returns TRUE if the button control on a dialog sheet will close the dialog box when pressed; otherwise, returns FALSE

61 Returns TRUE if the button control on a dialog sheet will be clicked when the user presses F1.

62 Returns the value of the control. For a check box or radio button, Returns 1 if it is selected, zero if it is not selected, or 2 if mixed. For a List box or dropdown box, returns the index number of the selected item, or zero if no item is selected. For a scroll bar, returns the numeric value of the scroll bar.

63 Returns the minimum value that a scroll bar or spinner button can have

64 Returns the maximum value that a scroll bar or spinner button can have

65 Returns the step increment value added or subtracted from the value of a scroll bar or spinner. This value is used when the arrow buttons are pressed on the control.

66 Returns the large, or "page" step increment value added or subtracted from the value of a scroll bar when it is clicked in the region between the thumb and the arrow buttons.

67 Returns the input type allowed in an edit box control:
1 = Text
2 = Integer
3 = Number (what type)
4 = Cell reference
5 = Formula

68 Returns TRUE if the edit box control allows multi-line editing with wrapped text; otherwise, it returns FALSE.

69 Returns TRUE if the edit box has a vertical scroll bar; otherwise, it returns FALSE.

70 Returns the object ID of the object that is linked to a list box or edit box. For a dropdown combo box that has an editable entry field, returns the object ID of itself. A dropdown box that can't be edited, returns FALSE.

71 Returns the number of entries in a List box, dropdown List box, or dropdown combo box.

72 Returns the text of the selected entry in a List box, dropdown List box, or dropdown combo box.

73 Returns the range used to fill the entries in a List box, dropdown List box, or dropdown combo box, as text. If an empty string is returned, then the control isn't filled from a range.

74 Returns the number of list lines displayed when a dropdown control is dropped.

75 Returns TRUE the object is displayed as 3-D; otherwise, it returns FALSE.

76 Returns the Far East phonetic accelerator key as text. Used for Far East versions of Microsoft Excel.

77 Returns the select status of the list box:
0 = single
1 = simple multi-select
2 = extended multi-select

78	Returns an array of TRUE and FALSE values indicating which items are selected in a list box. If TRUE, the item is selected; If FALSE, the item is not selected.
79	Returns TRUE if the add indent attribute is on for alignment. Returns FALSE if the add indent attribute is off for alignment. Used for only Far East versions of Microsoft Excel.

Object_id_text is the name and number, or number alone, of the object you want information about. Object_id_text is the text displayed in the reference area when the object is selected. If object_id_text is omitted, it is assumed to be the selected object. If object_id_text is omitted and no object is selected, GET.OBJECT returns the #REF! error value and interrupts the macro.

Start_num is the number of the first character in the text box or button or the first vertex in a polygon you want information about. Start_num is ignored unless a text box, button, or polygon is specified by type_num and object_id_text. If start_num is omitted, it is assumed to be 1.

Count_num is the number of characters in a text box or button, or the number of vertices in a polygon, starting at start_num, that you want information about. Count_num is ignored unless a text box, button, or polygon is specified by type_num and object_id_text. If count_num is omitted, it is assumed to be 255.

Item_index is the index number or position of the item in the list box or drop-down box that you want information about, ranging from 1 to the number of items in the list box or drop-down box.

Tip Use GET.OBJECT(45) - GET.OBJECT(43) to determine the height of an object and GET.OBJECT(44) - GET.OBJECT(42) to determine the width.

Examples

The following macro formula returns the reference of the cell under the upper-left corner of the object Oval 3 (assume the cell is E2):

`GET.OBJECT(4, "Oval 3")` returns "R2C5"

The following macro formula changes the protection status of the object Rectangle 2 if it is locked:

```
IF(GET.OBJECT(2, "Rectangle 2"), OBJECT.PROTECTION(FALSE))
```

The following macro formula returns characters 25 through 185 from the object Text 5:

```
GET.OBJECT(12, "Text 5", 25, 160)
```

Related Functions

CREATE.OBJECT Creates an object

FONT.PROPERTIES Applies a font to the selection

OBJECT.PROTECTION Controls how an object is protected

PLACEMENT Determines an object's relationship to underlying cells

GET.PIVOT.FIELD

Returns information about a field in a PivotTable report.

Syntax

GET.PIVOT.FIELD(type_num, pivot_field_name, pivot_table_name)

Type_num is a value from 1 to 17 that returns the following types of information:

Type_num	Value
1	Returns an array of all the items which make up pivot_field_name. The array is made up of text constants, dates or numbers depending on the field.
2	Returns an array of all items which are set to show with the pivot_field_name. The array is made up of text constants, dates or numbers depending on the field. The array is returned in the order that the items are displayed in the PivotTable report. If pivot_field_name is a page field, then the array contains only one element, the value corresponding to the active page (this could be all if the All item is showing).
3	Returns an array of all items which are hidden in the pivot_field_name. The array is made up of text constants, dates or numbers depending on the field. If pivot_field_name is a data field or the data header name, this function returns the #N/A! error value.
4	Returns an integer describing where the field is displayed in the active PivotTable report (either row or column): 0 = Hidden 1 = Row 2 = Col 3 = Page 4 = Data
5	Returns an array of all items in pivot_field_name that are group parents. The array is made up of text constants, dates or numbers depending on the field. The array is returned in the order which these items appear in the PivotTable report. Returns #N/A if there are no group parents and if the pivot_field_name is a data field or the data field header.

6 Returns a number between 0 and 4095 which describes the subtotals attached to the field. The number is the sum of the values associated with each subtotal function. See PIVOT.FIELD.PROPERTIES for a list of all the values associated with subtotal calculations. If the field is showing as a data field or data field header, #N/A! is returned.

7 Returns an integer describing the type of data contained in the field:

0 = Text

1 = Number

2 = Date

8 Returns an array five columns wide and one row high describing the summary function's custom calculation shown with the specified field (Data field) in the PivotTable report. The array will look as follows: {function, calculation, base field, base item, number format}. If pivot_field_name is not showing in the active PivotTable report as a data field, #N/A! is returned.

9 Returns a reference to all of pivot_field_name's items currently showing in the active PivotTable report. If pivot_field_name is hidden, #N/A! is returned. If pivot_field_name is a page field, the reference to the currently showing page item is returned. If pivot_field_name is a data field, a reference to all the data for this field in the PivotTable report is returned. The references are returned as text.

10 Returns a reference to the header cell for pivot_field_name. If pivot_field_name is a data field, a reference to all the headers in the data row or column is returned. If pivot_field_name is hidden, #N/A! is returned. The reference is returned as text.

11 Returns the number of grouped fields in the grouped field set which includes pivot_field_name. If pivot_field_name is neither a parent field nor a child field, 1 is returned. If pivot_field_name is a data field or data header name, the function returns the #N/A! error value.

12 Returns the level of pivot_field_name in the grouped field set which includes pivot_field_name. Returns 1 for the highest level parent field, 2 for its child field, and so on. If pivot_field_name is neither a parent field nor a child field, 1 is returned. If pivot_field_name is a data field or data header name, the function returns the #N/A! error value.

13 Returns the name of the parent field for pivot_field_name as a text constant. If pivot_field_name is not a child field, #N/A! is returned.

14 Returns the name of the child field for pivot_field_name as a text constant. If pivot_field_name is not a parent field, #N/A! is returned.

15	Returns a text constant representing the original name of the field in the data source.
16	Returns the position of the field among all the other fields in its orientation. For instance, a 1 would be returned if the field was the first row field.
17	Returns an array of all items in pivot_field_name that are group children. The array is made up of text constants, dates or numbers depending on the field. The array is returned in the order which these items appear in the PivotTable report. Returns #N/A if there are no group children, and if the pivot_field_name is a data field or the data field header.

Pivot_field_name is the name of the field that you want information about. If there is no field named pivot_field_name in the PivotTable report, returns #VALUE!.

Pivot_table_name is the name of a PivotTable report containing the field that you want information about. If omitted, the PivotTable report containing the active cell is used. If the active cell is not in a PivotTable report, the #VALUE! error value is returned.

Related Functions

GET.PIVOT.ITEM Returns information about an item in a PivotTable report.

GET.PIVOT.TABLE Returns information about a PivotTable report.

GET.PIVOT.ITEM

Returns information about an item in a PivotTable report.

Syntax

GET.PIVOT.ITEM(**type_num**, pivot_item_name, pivot_field_name, pivot_table_name)

Type_num is a value from 1 to 9 the represents the type of information you want about an item in a PivotTable report.

Type_num	Information
1	Returns the position of the item in its field. Returns #N/A if pivot_field_name is a data field. Returns #N/A! if the item is hidden.
2	Returns the reference to all the cells in the PivotTable header currently containing pivot_item_name. This reference is returned as text. If pivot_item_name is currently not showing in the PivotTable report, #N/A! is returned.

3	Returns the reference to all the data in the PivotTable report which is qualified by pivot_item_name. This reference is returned as text. If pivot_item_name is currently not showing in the PivotTable report, #N/A! is returned.
4	Returns an array of text constants representing the children of pivot_item_name if pivot_item_name is a parent. Otherwise the function returns #N/A!.
5	Returns a text constant representing the parent of pivot_item_name, if pivot_item_name exists as part of a group. Otherwise the function returns #N/A!.
6	Returns TRUE if pivot_item_name is a member of a group which is currently expanded to show detail. Returns FALSE if pivot_item_name is a member of a group currently collapsed to hide detail. If pivot_item_name is not a member of a group, the function returns #N/A!.
7	Returns TRUE if pivot_item_name is expanded to show detail. Returns FALSE if pivot_item_name is collapsed to hide detail.
8	Returns TRUE if the item pivot_item_name is currently visible, FALSE if it is hidden.
9	Returns the name of the item as it appeared in the original at a source. This will differ from the current item name only if the user changes the name of the item after creating the PivotTable report.

Pivot_item_name is the name of the item that you want information about. If there is no item named pivot_item_name in the PivotTable report, returns #VALUE!.

Pivot_field_name is the name of the field that you want information about. If there is no field named pivot_field_name in the PivotTable report, returns #VALUE!.

Pivot_table_name is the name of a PivotTable report containing the field that you want information about. If omitted, uses the PivotTable report containing the active cell. If the active cell is not in a PivotTable report, the #VALUE! error value is returned.

Related Functions

GET.PIVOT.FIELD Returns information about an item in a PivotTable report.

GET.PIVOT.TABLE Returns information about a PivotTable report.

GET.PIVOT.TABLE

Returns information about a PivotTable report.

Syntax

GET.PIVOT.TABLE(**type_num**,pivot_table_name)

Type_num is a value from 1 to 22 that represents a type of information you want about a PivotTable report.

Type_num	**Information**
1	Returns the name of the person who last updated the PivotTable report, as a text constant.
2	Returns the date the PivotTable report was last updated, as a serial number.
3	Returns a horizontal array of text constants representing all the fields in the PivotTable report.
4	Returns an integer representing the number of fields in the PivotTable report.
5	Returns a horizontal array of text constants representing all the visible fields in the PivotTable report (rows, columns, pages or data)
6	Returns a horizontal array of text constants representing all the hidden fields in the PivotTable report. Return #N/A if no hidden fields.
7	Returns a horizontal array of text constants representing the names of all the fields currently showing in the PivotTable report as row fields. Returns #N/A if there are no row fields.
8	Returns a horizontal array of text constants representing all the fields currently showing in the PivotTable report as column fields. Returns #N/A if no column fields exist.

9	Returns a horizontal array of text constants representing all the fields currently showing in the PivotTable report as page fields. Return #N/A if no page fields exist.
10	Returns a horizontal array of text constants representing all the fields currently showing in the PivotTable report as data fields. Returns #N/A if there are no data fields.
11	Returns the smallest rectangular reference which bounds the PivotTable report and all headers (not including the page header). This reference is returned as text.
12	Returns the smallest rectangular reference which bounds the PivotTable report and all headers (including the page headers). This reference is returned as text.
13	Returns the reference to the row header area as text. The row header area includes each row field header along with all the items in each row field. Returns #N/A if there are no row headers.
14	Returns the reference to the column header area as text. The column header area includes each column field header along with all the items in each column field. Returns #N/A if there are no column headers.
15	Returns the reference to the data header area as text. The data header area includes the data field header along with all the headers in the data row/col. Returns #N/A if there is no data field.
16	Returns a reference to all the page headers as text.
17	Returns the reference to the PivotTable report data area as text.
18	Returns TRUE if the PivotTable report is set to show row grand totals.
19	Returns TRUE if the PivotTable report is set to show column grand totals.

20	Returns TRUE if the user is saving data with the PivotTable report.
21	Returns TRUE if the PivotTable report is set up to Autoformat on pivoting.
22	Returns the data source of the PivotTable report. The kind of information returned depends on the data source: If the data source is a Microsoft Excel list or database, the cell reference is returned as text. If the data source is an external data source, then an array is returned. Each row consists of a SQL connection string with the remaining elements as the query string broken down into 200 character segments. If the data source is Multiple Consolidation ranges, then a two dimensional array is returned, each row of which consists of a reference and associated page field items. If the data source is another PivotTable report, then one of the above three kinds of information is returned.

Pivot_table_name is the name of a PivotTable report containing the field that you want information about. If omitted, uses the PivotTable report containing the active cell.

Remarks

Returns #VALUE! error value when pivot_table_name is not a valid PivotTable name on the active sheet and the active cell is not within a PivotTable report.

Related Functions

GET.PIVOT.FIELD Returns information about an item in a PivotTable report.

GET.PIVOT.ITEM Returns information about a PivotTable report.

GET.TOOL

Returns information about a button or buttons on a toolbar. Use GET.TOOL to get information about a button to use with functions that add, delete, or alter buttons.

Syntax

GET.TOOL(**type_num, bar_id, position**)

Type_num specifies what type of information you want GET.TOOL to return.

Type_num	Returns
1	The button's ID number. Gaps are represented by zeros.
2	The reference of the macro assigned to the button. If no macro is assigned, GET.TOOL returns the #N/A error value.
3	If the button is down, returns TRUE. If the button is up, returns FALSE.
4	If the button is enabled, returns TRUE. If the button is disabled, returns FALSE.
5	A logical value indicating the type of the face on the button: TRUE = bitmap FALSE = a default button face
6	The help_text reference associated with the custom button. If the button is built-in, returns #N/A.
7	The balloon_text reference associated with the custom button. If the button is built-in, returns the #N/A error value.
8	The Help context string associated with the custom button.
9	The Tip_text associated with the custom button.

Bar_id specifies the number or name of the toolbar for which you want information. For detailed information about bar_id, see ADD.TOOL.

Position specifies the position of the button on the toolbar. Position starts with 1 at the left side (if horizontal) or at the top (if vertical). A position can be occupied by a button or a gap.

Example

The following macro formula requests the help text associated with the third button in Toolbar2:

```
GET.TOOL(6, "Toolbar2", 3)
```

Related Functions

ADD.TOOL	Adds one or more buttons to a toolbar
DELETE.TOOL	Deletes a button from a toolbar
ENABLE.TOOL	Enables or disables a button on a toolbar
GET.TOOLBAR	Retrieves information about a toolbar

GET.TOOLBAR

Returns information about one toolbar or all toolbars. Use GET.TOOLBAR to get information about a toolbar to use with functions that add, delete, or alter toolbars.

Syntax

GET.TOOLBAR(**type_num**, bar_id)

Type_num specifies what type of information to return. If type_num is 8 or 9, GET.TOOLBAR returns an array of names or numbers of all visible or hidden toolbars. Otherwise, bar_id is required, and GET.TOOLBAR returns the requested information about the specified toolbar.

Type_num	Returns
1	A horizontal array of all tool IDs on the toolbar, ordered by position. Gaps are represented by zeros.
2	Number indicating the horizontal position (x-coordinate) of the toolbar in the docked or floating region. For more information, see SHOW.TOOLBAR.
3	Number indicating the vertical position (y-coordinate) of the toolbar in the docked or floating region.
4	Number indicating the width of the toolbar in points.
5	Number indicating the height of the toolbar in points.
6	Number indicating the toolbar location:

	1 = Top dock in the workspace
	2 = Left dock in the workspace
	3 = Right dock in the workspace
	4 = Bottom dock in the workspace
	5 = Floating
7	If the toolbar is visible, returns TRUE. If the toolbar is hidden, returns FALSE.
8	An array of toolbar IDs (names or numbers in the bar_id array) for all toolbars, visible and hidden.
9	An array of toolbar IDs (names or numbers in the bar_id array) for all visible toolbars.
10	If the toolbar is visible in full-screen mode, returns TRUE; otherwise, returns FALSE.

Bar_id specifies the number or name of a toolbar for which you want information. If type_num is 8 or 9, Microsoft Excel ignores bar_id. For detailed information about bar_id, see ADD.TOOL.

Remarks

If you request position information for a hidden toolbar, Microsoft Excel returns the position where the toolbar would appear if shown.

Examples

The following macro formula returns information about the width of Toolbar1:

```
GET.TOOLBAR(4, "Toolbar1")
```

When the following macro formula is entered as an array with CTRL+SHIFT+ENTER, the IDs of all visible toolbars are returned, and the array is named All_Bar_Ids:

```
SET.NAME("All_Bar_Ids", GET.TOOLBAR(9))
```

Related Functions

ADD.TOOL Adds one or more buttons to a toolbar

ADD.TOOLBAR Creates a new toolbar with the specified tools

DELETE.TOOLBAR Deletes custom toolbars

GET.TOOL Returns information about a tool or tools on a toolbar

SHOW.TOOLBAR Hides or displays a toolbar

GET.WINDOW

Returns information about a window. Use GET.WINDOW in a macro that requires the status of a window, such as its name, size, position, and display options.

Syntax

GET.WINDOW(**type_num**, window_text)

Type_num is a number that specifies what type of window information you want. The following list shows the possible values of type_num and the corresponding results:

Type_num	**Returns**
1	Name of the workbook and sheet in the window as text. For compatibility with Microsoft Excel version 4.0, if the window contains only a single sheet that has the same name as the workbook without its extension, returns only the name of the book. Otherwise, returns the name of the sheet in the form "[Book1]Sheet1".
2	Number of the window.
3	X position, measured in points from the left edge of the workspace (in Microsoft Excel for Windows) or screen (in Microsoft Excel for the Macintosh) to the left edge of the window.
4	Y position, measured in points from the bottom edge of the formula bar to the top edge of the window.
5	Width, measured in points.
6	Height, measured in points.
7	If window is hidden, returns TRUE; otherwise, returns FALSE.

The rest of the values for type_num apply only to worksheets and macro sheets, except where indicated:

Type_num	Returns
8	If formulas are displayed, returns TRUE; otherwise, returns FALSE.
9	If gridlines are displayed, returns TRUE; otherwise, returns FALSE.
10	If row and column headings are displayed, returns TRUE; otherwise, returns FALSE.
11	If zeros are displayed, returns TRUE; otherwise, returns FALSE.
12	Gridline and heading color as a number in the range 1 to 56, corresponding to the colors in the View tab of the Options dialog box; if color is automatic, returns 0.

Values 13 to 16 for type_num return arrays that specify which rows or columns are at the top and left edges of the panes in the window and the widths and heights of those panes.

- The first number in the array corresponds to the first pane, the second number to the second pane, and so on.
- If the edge of the pane occurs at the boundary between rows or columns, the number returned is an integer.
- If the edge of the pane occurs within a row or column, the number returned has a fractional part that represents the fraction of the row or column visible within the pane.
- The numbers can be used as arguments to the SPLIT function to split a window at specific locations.

Type_num	Returns
13	Leftmost column number of each pane, in a horizontal numeric array
14	Top row number of each pane, in a horizontal numeric array.
15	Number of columns in each pane, in a horizontal numeric array.

16	Number of rows in each pane, in a horizontal numeric array.
17	Number indicating the active pane: 1 = Upper, left, or upper-left 2 = Right or upper-right 3 = Lower or lower-left 4 = Lower-right
18	If window has a vertical split, returns TRUE; otherwise, returns FALSE.
19	If window has a horizontal split, returns TRUE; otherwise, returns FALSE.
20	If window is maximized, returns TRUE; otherwise, returns FALSE.
21	Reserved
22	If the Outline Symbols check box is selected in the View tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
23	Number indicating the size of the window (including charts): 1 = Restored 2 = Minimized (displayed as an icon) 3 = Maximized
24	If panes are frozen on the active window, returns TRUE; otherwise, returns FALSE.
25	The numeric magnification of the active window (as a percentage of normal size) as set in the Zoom dialog box, or 100 if none is specified.
26	Returns TRUE if horizontal scrollbars are displayed in the active window; otherwise, returns FALSE.
27	Returns TRUE if vertical scrollbars are displayed in the active window; otherwise, returns FALSE.
28	Returns the tab ratio of workbook tabs to horizontal scrollbar, from 0 to 1. The default is .6.

29	Returns TRUE if workbook tabs are displayed in the active window; otherwise, returns FALSE.
30	Returns the title of the active sheet in the window in the form "[Book1]Sheet1".
31	Returns the name of a workbook only, without read/write indicated. For example, if Book1.xls is read only, then "Book.xls" will be returned without "[Read Only]" appended.

Window_text is the name that appears in the title bar of the window that you want information about. If window_text is omitted, it is assumed to be the active window.

Examples

If the active window contains the workbook Book1, then:

`GET.WINDOW(1)` equals "Book1"

If the title of the active window is Macro1:3, then:

`GET.WINDOW(2)` equals 3

In Microsoft Excel for Windows, the following macro formula returns the gridline and heading color of REPORT.XLS:

```
GET.WINDOW(12, "REPORT.XLS")
```

In Microsoft Excel for the Macintosh, the following macro formula returns the gridline and heading color of REPORT MASTER:

```
GET.WINDOW(12, "REPORT MASTER")
```

Related Functions

GET.DOCUMENT Returns information about a workbook

GET.WORKSPACE Returns information about the workspace

GET.WORKBOOK

Returns information about a workbook.

Syntax

GET.WORKBOOK(type_num, name_text)

Type_num is a number that specifies what type of workbook information you want.

Type_num	Returns
1	The names of all sheets in the workbook, as a horizontal array of text values. Names are returned as [book]sheet.
2	This will always return the #N/A error value.
3	The names of the currently selected sheets in the workbook, as a horizontal array of text values.
4	The number of sheets in the workbook.
5	TRUE if the workbook has a routing slip; otherwise, FALSE.
6	The names of all of the workbook routing recipients who have not received the workbook, as a horizontal array of text values.
7	The subject line for the current routing slip, as text.
8	The message text for the routing slip, as text.
9	If the workbook is to be routed to recipients one after another, returns 1. If it is to be routed all at once, returns 2.
10	TRUE, if the Return When Done check box in the Routing Slip dialog box is selected; otherwise, FALSE.
11	TRUE, if the current recipient has already forwarded the current workbook; otherwise, FALSE.
12	TRUE, if the Track Status checkbox in the Routing Slip dialog box is selected; otherwise, FALSE.
13	Status of the workbook routing slip: 0 = Unrouted

	1 = Routing in progress, or the workbook has been routed to a user 2 = Routing is finished
14	TRUE, if the workbook structure is protected; otherwise, FALSE.
15	TRUE, if the workbook windows are protected; otherwise, FALSE.
16	Name of the workbook as text. The workbook name does not include the drive, directory or folder, or window number.
17	TRUE if the workbook is read only; otherwise, FALSE. This is the equivalent of GET.DOCUMENT(34).
18	TRUE if sheet is write-reserved; otherwise, FALSE. This is the equivalent of GET.DOCUMENT(35).
19	Name of the user with current write permission for the workbook. This is the equivalent of GET.DOCUMENT(36).
20	Number corresponding to the file type of the document as displayed in the Save As dialog box. This is the equivalent of GET.DOCUMENT(37).
21	TRUE if the Always Create Backup check box is selected in the Save Options dialog box; otherwise, FALSE. This is the equivalent of GET.DOCUMENT(40).
22	TRUE if the Save External Link Values check box is selected in the Calculation tab of the Options dialog box. This is the equivalent of GET.DOCUMENT(43).
23	TRUE if the workbook has a PowerTalk mailer; otherwise, FALSE. Returns #N/A if no OCE mailer is installed.
24	TRUE if changes have been made to the workbook since the last time it was saved; FALSE if book is unchanged (or when closed, will not prompt to be saved).
25	The recipients on the To line of a PowerTalk mailer, as a horizontal array of text.
26	The recipients on the Cc line of a PowerTalk mailer, as a horizontal array of text.
27	The recipients on the Bcc line of a PowerTalk mailer, as a horizontal array of text.

28	The subject of the PowerTalk mailer, as text.
29	The enclosures of the PowerTalk mailer, as a horizontal array of text.
30	TRUE, if the PowerTalk mailer has been received from another user (as opposed to just being added but not sent). FALSE, if the mailer has not been received from another user.
31	The date and time the PowerTalk mailer was sent, as a serial number. Returns the #N/A error value if the mailer has not yet been sent.
32	The sender name of the PowerTalk mailer, as text. Returns the #N/A error value if the mailer has not yet been sent.
33	The title of the workbook as displayed on the Summary tab of the Properties dialog box, as text.
34	The subject of the workbook as displayed on the Summary tab of the Properties dialog box, as text.
35	The author of the workbook as displayed on the Summary tab of the Properties dialog box, as text.
36	The keywords for the workbook as displayed on the Summary tab of the Properties dialog box, as text.
37	The comments for the workbook as displayed on the Summary tab of the Properties dialog box, as text.
38	The name of the active sheet.

Name_text   is the name of an open workbook. If name_text is omitted, it is assumed to be the active workbook.

Example

The following macro formula returns the name of the active sheet in the workbook named SALES.XLS:

```
GET.WORKBOOK(38, "SALES.XLS")
```

Related Functions

GET.DOCUMENT   Returns information about a workbook

WORKBOOK.SELECT   Selects the specified documents in a workbook

GET.WORKSPACE

Returns information about the workspace. Use GET.WORKSPACE in a macro that depends on the status of the workspace, such as the environment, version number, and available memory.

Syntax

GET.WORKSPACE(**type_num**)

Type_num is a number specifying the type of workspace information you want. The following list shows the type_num values and their corresponding results.

Type_num	**Returns**
1	Name of the environment in which Microsoft Excel is running, as text, followed by the environment's version number.
2	The version number of Microsoft Excel, as text (for example, "5.0").
3	If fixed decimals are set, returns the number of decimals; otherwise, returns 0.
4	If in R1C1 mode, returns TRUE; if in A1 mode, returns FALSE.
5	If scroll bars are displayed, returns TRUE; otherwise, returns FALSE. See also GET.WINDOW(26) and GET.WINDOW(27).
6	If the status bar is displayed, returns TRUE; otherwise, returns FALSE.
7	If the formula bar is displayed, returns TRUE; otherwise, returns FALSE.
8	If remote DDE requests are enabled, returns TRUE; otherwise, returns FALSE.
9	Returns the alternate menu key as text; if no alternate menu key is set, returns the #N/A error value.
10	Number indicating special modes: 1 = Data Find 2 = Copy

	3 = Cut 4 = Data Entry 5 = Unused 6 = Copy and Data Entry 7 = Cut and Data Entry If no special mode is set, returns 0.
11	X position of the Microsoft Excel workspace window, measured in points from the left edge of the screen to the left edge of the window. In Microsoft Excel for the Macintosh, always returns 0.
12	Y position of the Microsoft Excel workspace window, measured in points from the top edge of the screen to the top edge of the window. In Microsoft Excel for the Macintosh, always returns 0.
13	Usable workspace width, in points.
14	Usable workspace height, in points.
15	Number indicating maximized or minimized status of Microsoft Excel: 1 = Neither 2 = Minimized 3 = Maximized Microsoft Excel for the Macintosh always returns 3.
16	Amount of memory free (in kilobytes).
17	Total memory available to Microsoft Excel (in kilobytes).
18	If a math coprocessor is present, returns TRUE; otherwise, returns FALSE.
19	If a mouse is present, returns TRUE; otherwise, returns FALSE. In Microsoft Excel for the Macintosh, always returns TRUE.
20	If a group is present in the workspace, returns a horizontal array of sheets in the group; otherwise returns the #N/A error value.

21	If the Standard toolbar is displayed, returns TRUE; otherwise, returns FALSE.
22	DDE-application-specific error code.
23	Full path of the default startup directory or folder.
24	Full path of the alternate startup directory or folder; returns the #N/A error value if no alternate path has been specified.
25	If Microsoft Excel is set for relative recording, returns TRUE; if set for absolute recording, returns FALSE.
26	Name of user.
27	Name of organization.
28	If Microsoft Excel menus are switched to by the transition menu or help key, returns 1; if Lotus 1-2-3 Help is switched to, returns 2.
29	If transition navigation keys are enabled, returns TRUE.
30	A nine-item horizontal array of global (default) print settings that can be set by the LINE.PRINT function: Setup text Left margin Right margin Top margin Bottom margin Page length Logical value indicating whether to wait after printing each page (TRUE) or use continuous form feeding (FALSE) Logical value indicating whether the printer has automatic line feeding (TRUE) or requires line feed characters (FALSE) The number of the printer port

31 If a currently running macro is in single step mode, returns TRUE; otherwise, returns FALSE.

32 The current location of Microsoft Excel as a complete path.

33 A horizontal array of the names in the New list, in the order they appear.

34 A horizontal array of template files (with complete paths) in the New list, in the order they appear (returns the names of custom template files and the #N/A error value for built-in document types).

35 If a macro is paused, returns TRUE; FALSE otherwise.

36 If the Allow Cell Drag And Drop check box is selected in the Edit tab of the Options dialog box that appears when you click the Options command on the Tools menu, returns TRUE; otherwise, returns FALSE.

37 A 45-item horizontal array of the items related to country versions and settings. Use the following macro formula to return a specific item, where number is a number in the list below:

```
INDEX(GET.WORKSPACE(37), number)
```

These values apply to country codes:
1 = Number corresponding to the country version of Microsoft Excel.
2 = Number corresponding to the current country setting in the Microsoft Windows Control Panel or the country number as determined by your Apple system software

These values apply to number separators:
3 = Decimal separator
4 = Zero (or 1000) separator
5 = List separator

These values apply to R1C1-style references:
6 = Row character
7 = Column character
8 = Lowercase row character
9 = Lowercase column character
10 = Character used instead of the left bracket ([)
11 = Character used instead of the right bracket (])

These values apply to array characters:
12 = Character used instead of the left bracket ({)
13 = Character used instead of the right bracket (})
14 = Column separator

15 = Row separator
16 = Alternate array item separator to use if the current array separator is the same as the decimal separator

These values apply to format code symbols:
17 = Date separator
18 = Time separator
19 = Year symbol
20 = Month symbol
21 = Day symbol
22 = Hour symbol
23 = Minute symbol
24 = Second symbol
25 = Currency symbol
26 = "General" symbol

These values apply to format codes:
27 = Number of decimal digits to use in currency formats
28 = Number indicating the current format for negative currencies:
  0 = ($currency) or (currency$)
  1 = -$currency or -currency$
  2 = $-currency or currency-$
  3 = $currency- or currency$-
where currency is any number and the $ represents the current currency symbol.
29 = Number of decimal digits to use in noncurrency number formats
30 = Number of characters to use in month names
31 = Number of characters to use in weekday names
32 = Number indicating the date order:
  0 = Month-Day-Year
  1 = Day-Month-Year
  2 = Year-Month-Day

These values apply to logical format values:
33 = TRUE if using 24-hour time; FALSE if using 12-hour time.
34 = TRUE if not displaying functions in English; otherwise, returns FALSE.
35 = TRUE if using the metric system; FALSE if using the English measurement system.
36 = TRUE if a space is added before the currency symbol; otherwise, returns FALSE.
37 = TRUE if currency symbol precedes currency values; FALSE if it follows currency values.
38 = TRUE if using minus sign for negative numbers; FALSE if using parentheses.
39 = TRUE if trailing zeros are displayed for zero currency values; otherwise, returns FALSE.

	40 = TRUE if leading zeros are displayed for zero currency values; otherwise, returns FALSE. 41 = TRUE if leading zero is displayed in months (when months are displayed as numbers); otherwise, returns FALSE. 42 = TRUE if leading zero is shown in days (when days are displayed as numbers); otherwise, returns FALSE. 43 = TRUE if using four-digit years; FALSE if using two-digit years. 44 = TRUE if date order is month-day-year when displaying dates in long form; FALSE if date order is day-month-year. 45 = TRUE if leading zero is shown in the time; otherwise, returns FALSE.
38	The number 0, 1, or 2 indicating the type of error-checking as set by the ERROR function. For more information, see ERROR.
39	A reference in R1C1-text form to the currently defined error-handling macro (set by the ERROR function), or the #N/A error value if none is specified.
40	If screen updating is turned on (set by the ECHO function), returns TRUE; otherwise, returns FALSE.
41	A horizontal array of cell ranges, as R1C1-style text, that were previously selected with the Go To command from the Edit menu or the FORMULA.GOTO macro function. If the book has multiple sheets, or if the single sheet in the workbook is named differently than the workbook itself, returns names as [Book]Sheet.
42	If your computer is capable of playing sounds, returns TRUE; otherwise, returns FALSE.
43	If your computer is capable of recording sounds, returns TRUE; otherwise, returns FALSE.
44	A three-column array of all currently registered procedures in dynamic link libraries (DLLs). The first column contains the names of the DLLs that contain the procedures (in Microsoft Excel for Windows) or the names of the files that contain the code resources (in Microsoft Excel for the Macintosh). The second column contains the names of the procedures in the DLLs (in Microsoft Excel for Windows) or code resources (in Microsoft Excel for the Macintosh). The third column contains text strings specifying the data types of the return values, and the number and data types of the arguments. For more information about DLLs and code resources and data types, see Using the CALL and REGISTER functions in Microsoft Excel Help.

45	If Microsoft Windows for Pen Computing is running, returns TRUE; otherwise, returns FALSE.
46	If the Move Selection After Enter check box is selected in the Edit tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
47	Reserved.
48	Path to the library subdirectory for Microsoft Excel, as text.
49	MAPI session currently in use, returned as a string of hex digits encoding the mail session value.
50	If the Full Screen mode is on, returns TRUE; otherwise, FALSE.
51	If the formula bar is displayed in Full Screen mode, returns TRUE; otherwise, FALSE.
52	If the status bar is displayed in Full Screen mode, returns TRUE; otherwise, FALSE.
53	The name of the topmost custom dialog sheet currently running in a modal window, or #N/A if no dialog sheet is currently running.
54	If the Edit Directly In Cell check box is selected on the Edit tab in the Options dialog box, returns TRUE; otherwise, returns FALSE.
55	TRUE if the Alert Before Overwriting Cells check box in the Edit tab on Options dialog box is selected; otherwise, FALSE.
56	Standard font name in the General tab in the Options dialog box, as text.
57	Standard font size in the General tab in the Options dialog box, as a number
58	If the Recently Used File list check box in the General tab on the Options dialog box is selected, returns TRUE; otherwise, FALSE.
59	If the Display Old Menus check box in the General tab on the Options dialog box is selected, returns TRUE; otherwise, FALSE.
60	If the Tip Wizard is enabled, returns TRUE; otherwise, FALSE.
61	Number of custom list entries listed in the Custom Lists tab of the Options dialog box.
62	Returns information about available file converters.

63	Returns the type of mail system in use by Excel: 0 = No mail transport detected 1 = MAPI based transport 2 = PowerTalk based transport (Macintosh only)
64	If the Ask To Update Automatic Links check box in the Edit tab of the Options dialog box is selected, returns TRUE; otherwise, FALSE.
65	If the Cut, Copy, And Sort Objects With Cells check box in the Edit tab on the Options dialog box is selected, returns TRUE; otherwise, FALSE.
66	Default number of sheets in a new workbook, as a number, from the General tab on Options dialog box.
67	Default file directory location, as text, from the General tab in the Options dialog box.
68	If the Show ScreenTips On Toolbars check box in the Options tab in the Customize dialog box is selected, returns TRUE; otherwise, FALSE.
69	If the Large Icons check box in the Options tab in the Customize dialog box is selected, returns TRUE; otherwise, FALSE.
70	If the Prompt For Workbook Properties check box in the General tab on the Options dialog box is selected, returns TRUE; otherwise, FALSE.
71	TRUE if Microsoft Excel is open for in-place object editing (OLE). If FALSE, it is opened normally.
72	TRUE if the Color Toolbars check box is selected in the Toolbars dialog box. FALSE if the Color Toolbars check box is not selected. This argument is for compatibility with Microsoft Excel version 5.0.

Related Functions

GET.DOCUMENT   Returns information about a workbook

GET.WINDOW   Returns information about a window

GOAL.SEEK

Equivalent to clicking the Goal Seek command on the Tools menu. Calculates the values necessary to achieve a specific goal. If the goal is an amount returned by a formula, the GOAL.SEEK function calculates values that, when supplied to your formula, cause your formula to return the amount you want.

Syntax

GOAL.SEEK(**target_cell, target_value, variable_cell**)

GOAL.SEEK?(target_cell, target_value, variable_cell)

Target_cell   corresponds to the Set Cell box in the Goal Seek dialog box and is a reference to the cell containing the formula. If target_cell does not contain a formula, Microsoft Excel displays an error message.

Target_value   corresponds to the To Value box in the Goal Seek dialog box and is the value you want the formula in target_cell to return. This value is called a goal.

Variable_cell   corresponds to the By Changing Cell box in the Goal Seek dialog box and is the single cell that you want Microsoft Excel to change so that the formula in target_cell returns target_value. Target_cell must depend on variable_cell; if it does not, Microsoft Excel will not be able to find a solution.

Remarks

The max_num and max_change values set with the CALCULATION function can be used to change the solution process. Max_num sets the number of iterations; max_change determines the precision of the solution.

Tip   You can also use Microsoft Excel Solver to help solve your math equations for optimal values.

Related Functions

Related functions include the SOLVER functions, such as SOLVER.OPTIONS, SOLVER.SOLVE, and so on.

GOTO

Directs a macro to continue running at the upper-left cell of reference. Use GOTO to direct macro execution to another cell or a named range.

Syntax

GOTO(**reference**)

Reference   is a cell reference or a name that is defined as a reference. Reference can be an external reference to another macro sheet. If that macro sheet is not open, GOTO displays a message.

Tip   It's often preferable to use IF, ELSE, ELSE.IF, and END.IF instead of GOTO when you want to perform multiple actions based on a condition because the IF method makes your macros more structured.

Examples

If A1 contains the #N/A error value, then when the following formula is calculated, the macro branches to C3:

```
IF(ISERROR($A$1), GOTO($C$3))
```

You can also use macro names with GOTO statements. The following macro formula branches macro execution to a macro named Compile:

```
GOTO(Compile)
```

Because Compile is a named range, it should not be enclosed in quotation marks.

Related Function

FORMULA.GOTO   Selects a named area or reference on any open workbook

GRIDLINES

Allows you to turn chart gridlines on and off.

Arguments are logical values corresponding to the check boxes in the Gridlines dialog box. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box. If omitted, the setting is not changed. If a chart is not active, produces a error and halts the macro.

Syntax

GRIDLINES(x_major, x_minor, y_major, y_minor, z_major, z_minor, 2D_effect)

GRIDLINES?(x_major, x_minor, y_major, y_minor, z_major, z_minor, 2D_effect)

X_major   corresponds to the Category (X) Axis: Major Gridlines check box.

X_minor   corresponds to the Category (X) Axis: Minor Gridlines check box.

Y_major   corresponds to the Value (Y) Axis: Major Gridlines check box. On 3-D charts, y_major corresponds to the Series (Y) Axis: Major Gridlines check box.

Y_minor   corresponds to the Value (Y) Axis: Minor Gridlines check box. On 3-D charts, y_minor corresponds to the Series (Y) Axis: Minor Gridlines check box.

Z_major   corresponds to the Value (Z) Axis: Major Gridlines check box (3-D only).

Z_minor   corresponds to the Value (Z) Axis: Minor Gridlines check box (3-D only).

2D_effect   corresponds to the 2-D Walls and Gridlines check box (3-D only).

GROUP

Creates a single object from several selected objects and returns the object identifier of the group (for example, "Group 5"). Use GROUP to combine a number of objects so that you can move or resize them together.

If no object is selected, only one object is selected, or a group is already selected, GROUP returns the #VALUE! error value and interrupts the macro.

Syntax

GROUP( )

Related Function

UNGROUP   Separates a grouped object

ECHO

Controls screen updating while a macro is running. If a large macro uses many commands that update the screen, use ECHO to make the macro run faster.

Syntax

ECHO(logical)

Logical is a logical value specifying whether screen updating is on or off.

- If logical is TRUE, Microsoft Excel selects screen updating.
- If logical is FALSE, Microsoft Excel clears screen updating.
- If logical is omitted, Microsoft Excel changes the current screen update condition.

Remarks

- Screen updating is always turned back on when a macro ends.
- You can use GET.WORKSPACE to determine whether screen updating is on or off.

Related Function

GET.WORKSPACE Returns information about the workspace

EDITBOX.PROPERTIES

Sets the properties of an edit box on a dialog sheet.

Syntax

EDITBOX.PROPERTIES(validation_num, multiline_logical, vscroll_logical, password_logical)

EDITBOX.PROPERTIES?(validation_num, multiline_logical, vscroll_logical, password_logical)

Validation_num is the validation applied to the edit box when the dialog is dismissed. If the edit box contains a value other than the type specified (or validation), an error is returned.

Validation_num	Type
1	Text
2	Integer
3	Number (allows floating point)

4	Reference
5	Formula

Multiline_logical is a logical value specifying whether word wrapping is allowed in the edit box control. If TRUE, word wrapping is allowed. If FALSE, word wrapping is not allowed

Vscroll_logical is a logical value specifying whether edit box displays a vertical scrollbar. If TRUE, a scrollbar is displayed. If FALSE, a scrollbar is not displayed.

Password_logical is a logical value specifying whether edit box displays characters as the user types. If TRUE, asterisks (*) are displayed as the user types. If FALSE, no asterisks are displayed.

Related Functions

CHECKBOX.PROPERTIES Sets various properties of check box and option box controls

PUSHBUTTON.PROPERTIES Sets the properties of the push button control

EDIT.COLOR

Equivalent to clicking the Modify button on the Color tab, which appears when you click the Options command on the Tools menu. Defines the color for one of the 56 color palette boxes.

Use EDIT.COLOR if you want to use a color that is not currently on the palette and if your system hardware has more than 56 colors available. After you set the color for the color box, any items previously formatted with that color are displayed in the new color.

Syntax

EDIT.COLOR(**color_num**, red_value, green_value, blue_value)

EDIT.COLOR?(**color_num**, red_value, green_value, blue_value)

Color_num is a number from 1 to 56 specifying one of the 56 color palette boxes for which you want to set the color.

Red_value, green_value, and blue_value are numbers that specify how much red, green, and blue are in each color.

- In Microsoft Excel for Windows, red_value, green_value, and blue_value are numbers from 0 to 255.
- In Microsoft Excel for the Macintosh, red_value, green_value, and blue_value are also numbers from 0 to 255. However, the color editing dialog box displays numbers from 0 to 65, 535. Microsoft Excel automatically converts the numbers between the two ranges. This allows you to display similar colors in all operating environments without modifying your macros.
- If red_value, green_value, and blue_value are all set to 255, the resulting color is white. If they are all set to zero, the resulting color is black.
- If red_value, green_value, or blue_value is omitted, Microsoft Excel assumes it to be the appropriate value for that color_num.

Remarks

- Your system hardware determines the number of unique colors that you can choose from and the number of colors that can be displayed on the screen at the same time.
- EDIT.COLOR does not use hue, saturation, or brightness values. If you are using the macro recorder and set the color of a color palette box using hue, saturation, and luminance, Microsoft Excel records the corresponding red, green, and blue values instead.
- The dialog-box form of this function, EDIT.COLOR?(color_num), displays your system's color editing dialog box. The default red_value, green_value, and blue_value are determined by the current settings for the color_num you specify. Color_num is a required argument for the dialog-box form of this function.

Related Function

COLOR.PALETTE   Copies a color palette from one workbook to another

EDIT.DELETE

Equivalent to clicking the Delete command on the Edit menu. Removes the selected cells from the worksheet and shifts other cells to close up the space.

Syntax

EDIT.DELETE(shift_num)

EDIT.DELETE?(shift_num)

Shift_num   is a number from 1 to 4 specifying whether to shift cells left or up after deleting the current selection or else to delete the entire row or column.

Shift_num	**Result**
1	Shifts cells left
2	Shifts cells up
3	Deletes entire row
4	Deletes entire column

- If shift_num is omitted and if one cell or a horizontal range is selected, EDIT.DELETE shifts cells up.
- If shift_num is omitted and a vertical range is selected, EDIT.DELETE shifts cells left.

Related Function

CLEAR Clears specified information from the selected cells or chart

EDITION.OPTIONS

Sets options in, or performs actions on, the specified publisher or subscriber. In Microsoft Excel for Windows, EDITION.OPTIONS also allows you to cancel a publisher or subscriber created in Microsoft Excel for the Macintosh.

Syntax

EDITION.OPTIONS(**edition_type**, edition_name, reference, **option**, appearance, size, formats)

Edition_type is the number 1 or 2 specifying the type of edition.

Edition_type	**Type of edition**
1	Publisher
2	Subscriber

Edition_name is the name of the edition you want to change the edition options for or to perform actions on. If edition_name is omitted, reference is required.

Reference specifies the range (given in text form as a name or an R1C1-style reference) occupied by the publisher or subscriber.

- Reference is required if you have more than one publisher or subscriber of edition_name on the active workbook. Use reference to specify the location of the publisher or subscriber for which you want to set options.
- If edition_type is 1 and the publisher is an embedded chart, or if edition_type is 2 and the subscriber is a picture, reference is the object identifier as displayed in the reference area.
- If reference is omitted, edition_name is required.

Option is a number from 1 to 6 specifying the edition option you want to set or the action you want to take, according to the following two tables. Options 2 to 6 are only available if you are using Microsoft Excel for the Macintosh with system software version 7.0 or later.

If a publisher is specified, then option applies as follows.

Option	Action
1	Cancels the publisher
2	Sends the edition now
3	Selects the range or object published to the specified edition
4	Automatically updates the edition when the file is saved
5	Updates the edition on request only
6	Changes the edition file as specified by appearance, size, and formats

If a subscriber is specified, then option applies as follows.

Option	Action
1	Cancels the subscriber
2	Gets the latest edition
3	Opens the publisher workbook
4	Automatically updates when new data is available
5	Update on request only

The following three arguments are available only when option is 6.

Appearance specifies whether the selection is published as shown on screen or as shown when printed. The default value for appearance is 1 if the selection is a sheet or macro sheet and 2 if the selection is a chart.

Appearance	Selection is published
1	As shown on screen
2	As shown when printed

Size specifies the size of a published chart. Size is only available if a chart is to be published.

Size	Chart size is published
1 or omitted	As shown on screen
2	As shown when printed

Formats is a number specifying the format of the file.

Formats	File format
1 or omitted	PICT
2	BIFF
4	RTF
8	VALU

You can also use the sum of the allowable file formats. For example, a value of 6 specifies BIFF and RTF.

Example

The following macro formula opens the workbook (and application) that published the edition named Monthly Totals:

```
EDITION.OPTIONS(2, "Monthly Totals", , 3)
```

Related Functions

CREATE.PUBLISHER	Creates a publisher from the selection

GET.LINK.INFO	Returns information about a link
SUBSCRIBE.TO	Inserts contents of an edition into the active workbook

EDIT.OBJECT

Equivalent to clicking the Edit command on the (selected object) Object submenu of the Edit menu. Starts the application associated with the selected object and makes the object available for editing or other actions.

Syntax

EDIT.OBJECT(verb_num)

Verb_num is a number specifying which verb to use while working with the object, that is, what you want to do with the object.

- The available verbs are determined by the object's source application. 1 often specifies "edit, " and 2 often specifies "play" (for sound, animation, and so on). For more information, consult the documentation for the object's application to see how it supports object linking and embedding (OLE).
- If the object does not support multiple verbs, verb_num is ignored.
- If verb_num is omitted, it is assumed to be 1.

Remarks

Your macro pauses while you're editing the object and resumes when you return to Microsoft Excel.

Related Function

INSERT.OBJECT	Creates an object of a specified type

EDIT.REPEAT

Equivalent to clicking the Repeat command on the Edit menu. Repeats certain actions and commands. EDIT.REPEAT is available in the same situations as the Repeat command.

Syntax

EDIT.REPEAT( )

EDIT.SERIES

Equivalent to clicking the Edit Series command on the Chart menu in Microsoft Excel version 4.0. Creates or changes chart series by adding a new SERIES formula or modifying an existing SERIES formula in the topmost chart type. Chart types are displayed in the following order from top to bottom: XY (Scatter), Line, Column, Bar, Area.

Syntax

EDIT.SERIES(**series_num**, **name_ref**, **x_ref**, **y_ref**, **z_ref**, **plot_order**)

EDIT.SERIES?(series_num, name_ref, x_ref, y_ref, z_ref, plot_order)

Series_num is the number of the series you want to change. If series_num is 0 or omitted, Microsoft Excel creates a new data series.

Name_ref is the name of the data series. It can be an external reference to a single cell, a name defined as a single cell, or a name defined as a sequence of characters. Name_ref can also be text (for example, "Projected Sales").

X_ref is an external reference to the name of the sheet and the cells that contain one of the following sets of data:

- Category labels for all charts except xy (scatter) charts
- X-coordinate data for xy (scatter) charts

Y_ref is an external reference to the name of the sheet and the cells that contain values (or y-coordinate data in xy (scatter) charts) for all 2-D charts. Y_ref is required in 2-D charts but does not apply to 3-D charts.

Z_ref is an external reference to the name of the sheet and the cells that contain values for all 3-D charts. Z_ref is required in 3-D charts but does not apply to 2-D charts.

Plot_order is a number specifying whether the data series is plotted first, second, and so on, in the chart type.

- If you assign a plot_order to a series, Microsoft Excel plots that series in the order you specify, and the series that previously had that plot order (and any series following it) has its plot order increased by one.
- If you add a series to a chart with an overlay, the number of series in the main chart does not change, so if the series is added to the main chart, then the series that was plotted last in the main chart will be plotted first in the overlay chart. To change which series is plotted first in the overlay chart, use the (chart type) Group command from the Format menu, and then select the Series Order tab in the Format (chart type) Group dialog box. You can also use the FORMAT.OVERLAY function.
- If you omit plot_order when you add a new series, then Microsoft Excel plots that series last and assigns it the correct plot_order value.
- The maximum value for plot_order is 255.

Remarks

To change where a series is plotted within a chart, you can change the chart type, using the FORMAT.CHART function, or the plot order. Plot order affects where the series appears within the chart type only.

X_ref, y_ref, and z_ref can be arrays or references to a nonadjacent selection, although they cannot be names that refer to a nonadjacent selection. If you specify a nonadjacent selection for any of these arguments, make sure to enclose the reference to the selection in parentheses so that Microsoft Excel does not treat the components of the references as separate arguments.

Tip To delete a data series, use the SELECT("Sn") macro function, where n is the series number, followed by the FORMULA("") macro function. You can also use the CLEAR function instead of FORMULA.

Related Function

FORMAT.CHART

EDIT.TOOL

Displays the Button Editor dialog box, which you use to change the appearance of a button on a toolbar.

Syntax

EDIT.TOOL(**bar_id**, **position**)

Bar_id is the number of the toolbar containing the button you want to edit. For a list of toolbar numbers, see ADD.TOOL. Use the GET.TOOLBAR function to return the information about a toolbar.

Position is the position on the toolbar of the button you want to edit. Buttons are numbered from the left starting at 1. Gaps between buttons are counted as positions.

Related Functions

ADD.TOOL Adds a button to a toolbar

GET.TOOLBAR Returns information about a toolbar

ELSE

Used with IF, ELSE.IF, and END.IF to control which functions are carried out in a macro. ELSE signals the beginning of a group of formulas in a macro sheet that will be carried out if the results of all preceding ELSE.IF statements and the preceding IF statement are FALSE. Use ELSE with IF, ELSE.IF, and END.IF when you want to perform multiple actions based on a condition. This method is preferable to using GOTO because it makes your macros more structured.

Syntax

ELSE( )

Remarks

ELSE must be entered in a cell by itself. In other words, the cell can contain only "=ELSE()".

For more information about ELSE, ELSE.IF, END.IF, and IF, and for examples of these functions, see form 2 of the IF function.

Related Functions

ELSE.IF Specifies an action to take if an IF or another ELSE.IF function returns FALSE

END.IF Ends a group of macro functions started with an IF statement

IF Specifies an action to take if a logical test is TRUE

ELSE.IF

Used with IF, ELSE, and END.IF to control which functions are carried out in a macro. ELSE.IF signals the beginning of a group of formulas in a macro sheet that will be carried out if the preceding IF or ELSE.IF function returns FALSE and if logical_test is TRUE. Use ELSE.IF with IF, ELSE, and END.IF when you want to perform multiple actions based on a condition. This method is preferable to using GOTO because it makes your macros more structured.

Syntax

ELSE.IF(**logical_test**)

Logical_test   is a logical value that ELSE.IF uses to determine what functions to carry out next—that is, where to branch.

- If logical_test is TRUE, Microsoft Excel carries out the functions between the ELSE.IF function and the next ELSE.IF, ELSE, or END.IF function.
- If logical_test is FALSE, Microsoft Excel immediately branches to the next ELSE.IF, ELSE, or END.IF function.

Remarks

- ELSE.IF must be entered in a cell by itself.
- Logical_test will always be evaluated, even if the ELSE.IF section is not reached (due to a previous IF or ELSE.IF logical_test evaluating to TRUE). For this reason, you should not use formulas that carry out actions for logical_test. If you need to base the ELSE.IF condition on the return value of a formula that carries out an action, use the form "ELSE, IF(logical_test), and END.IF" in place of "ELSE.IF(logical_test)."

For more information about ELSE, ELSE.IF, END.IF, and IF, and for examples of these functions, see form 2 of the IF function.

Related Functions

ELSE   Specifies an action to take if an IF function returns FALSE

END.IF   Ends a group of macro functions started with an IF statement

IF   Specifies an action to take if a logical test is TRUE

EMBED

Displayed in the formula bar when an embedded object is selected. EMBED cannot be entered on a sheet or used in a macro.

Syntax

EMBED(**object_class**, item)

Object_class   is the name of the application and document type that created the embedded object. For example, the object_class arguments used when Microsoft Excel sheets are embedded in other applications are "Excel.sheet.5" and "Excel.Chart.5".

Item   is the area selected to copy, and determines the view on the embedded document. When item is empty text (""), EMBED creates a view on the entire document.

Remarks

If you delete the EMBED formula, the embedded object remains on the sheet as a graphic, and the link to the creating application is deleted. Double-clicking the object no longer starts the creating application.

ENABLE.COMMAND

Enables or disables a custom command or menu. Disabled commands appear dimmed and can't be chosen. Use ENABLE.COMMAND to control which commands the user can click in a menu bar.

Syntax

ENABLE.COMMAND(**bar_num**, **menu**, **command**, **enable**, subcommand)

Bar_num is the menu bar in which a command resides. Bar_num can be the number of a built-in menu bar or the number returned by a previously run ADD.BAR function. See ADD.COMMAND for a list of the built-in menu bar numbers.

Menu is the menu on which the command resides. Menu can be either the name of a menu as text or the number of a menu. Menus are numbered starting with 1 from the left of the screen.

Command is the command you want to enable or disable. Command can be either the name of the command as text or the number of the command. The top command on a menu is command 1. If command is 0, ENABLE.COMMAND enables or disables the entire menu.

Enable is a logical value specifying whether the command should be enabled or disabled. If enable is TRUE, Microsoft Excel enables the command; if FALSE, it disables the command.

Subcommand is the name of the command on a submenu that you want to enable. If you use subcommand, you must use command as the name of the submenu. Use subcommand 0 to enable an entire submenu.

Remarks

- You cannot disable built-in commands. If the specified command is a built-in command or does not exist, ENABLE.COMMAND returns the #VALUE! error value and interrupts the macro.
- You can hide any shortcut menu from users by using ENABLE.COMMAND with command set to 0.

Example

The following macro formula disables a custom command that had been added previously to the View menu on the worksheet and macro sheet menu bar:

```
ENABLE.COMMAND(10, "View", "Audit...", FALSE)
```

Related Functions

ADD.BAR	Adds a menu bar
ADD.COMMAND	Adds a command to a menu
CHECK.COMMAND	Adds or deletes a check mark to or from a command
DELETE.COMMAND	Deletes a command from a menu
RENAME.COMMAND	Changes the name of a command or menu

ENABLE.OBJECT

Enables or disables a drawing object or the selected drawing object. A disabled object will not run any macro events assigned to it, and the controls will be grayed out.

Syntax

ENABLE.OBJECT(object_id_text, **enable_logical**)

Object_id_text is the name of the object(s) as text. If omitted, the selected object(s) are assumed.

Enable_logical is a logical value that specifies whether the object is to be enabled. If TRUE, the object is enabled. If FALSE, the object is disabled.

Examples

`ENABLE.OBJECT("Button 2",FALSE)` disables the button with object name Button 2 on the dialog box.

Related Function

SET.CONTROL.VALUE Changes the value of the active control

ENABLE.TIPWIZARD

This function should not be used. The TipWizard has been removed from Microsoft Excel.

ENABLE.TOOL

Enables or disables a button on a toolbar. An enabled button can be accessed by the user. Disabled buttons may still be visible but cannot be accessed. Use ENABLE.TOOL to control which buttons the user can click in a particular situation.

Syntax

ENABLE.TOOL(**bar_id, position**, enable)

Bar_id is the number or name of a toolbar on which the button resides. For detailed information about bar_id, see ADD.TOOL.

Position specifies the position of the button on the toolbar. Position starts with 1 at the left side (if horizontal) or from the top (if vertical).

Enable specifies whether the button can be accessed. If enable is TRUE or omitted, the user can access the button; if FALSE, the user cannot access it.

Remarks

Microsoft Excel sounds a tone if you click a disabled button.

Example

The following macro formula enables the fourth button in Toolbar1:

```
ENABLE.TOOL("Toolbar1", 4, TRUE)
```

Related Function

GET.TOOL   Returns information about a button or buttons on a toolbar

END.IF

Ends a block of functions associated with the preceding IF function. You must include one and only one END.IF function for each macro-sheets-only syntax form (syntax 2) of the IF function in a macro. Syntax 1 of the IF function, which can be used on both worksheets and macro sheets, does not require an END.IF function. Use END.IF with IF, ELSE, and ELSE.IF when you want to perform multiple actions based on a condition. This method is preferable to using GOTO because it makes your macros more structured.

Syntax

END.IF( )

Remarks

- If you accidentally omit an END.IF function, your macro will end with an error at the cell containing the first IF function that does not have a corresponding END.IF function.
- END.IF must be entered in a cell by itself.
- For more information about ELSE, ELSE.IF, END.IF, and IF, and for examples of these functions, see form 2 of the IF function.

Related Functions

ELSE   Specifies an action to take if an IF function returns FALSE

ELSE.IF   Specifies an action to take if an IF or another ELSE.IF function returns FALSE

IF   Specifies an action to take if a logical test is TRUE

ENTER.DATA

Turns on Data Entry mode and allows you to select and to enter data into the unlocked cells in the current selection only (the data entry area). Use ENTER.DATA when you want to enter data only in a specific part of your sheet. You can then use that part of the sheet as a simple data form.

Syntax

ENTER.DATA(logical)

Logical   is a logical value that turns Data Entry mode on or off.

- If logical is TRUE, Data Entry mode is turned on; if FALSE, Data Entry mode is turned off and data entry, cell movement, and cell selection return to normal. If logical is omitted, ENTER.DATA toggles Data Entry mode.
- Logical can also be the number 2. This setting turns on Data Entry mode and prevents the ESC key from turning it off.

- Logical can also be a reference. Using a reference for this argument turns on Data Entry mode for the supplied reference.

Remarks

- In Data Entry mode, you can move the active cell and select cell ranges only in the data entry area. The arrow keys and the TAB and SHIFT+TAB keys move from one unlocked cell to the next. The HOME and END keys move to the first and last cell in the data entry area, respectively. You cannot select entire rows or columns, and clicking a cell outside the data entry area does not select it.
- The only commands available in Data Entry mode are commands normally available to protected workbooks.
- To turn off Data Entry mode, press ESC (unless logical is 2), activate another sheet in the active workbook window, or use another ENTER.DATA function. If you use another ENTER.DATA function, you will usually design your macros in one of two ways:
    - The macro turns on Data Entry mode, pauses while you enter data, resumes, and then turns off Data Entry mode.
    - The macro turns on Data Entry mode and ends. After entering data, another macro turns off Data Entry mode; this latter macro could be assigned to a "Finished" button, for example.

  With either method, you can use Microsoft Excel's ON functions to resume or run other macros based on an event, such as pressing the CONTROL+D keys.

Tips

- Normally you use Data Entry mode to enter data, but you can also prevent someone from entering data or moving the active cell by locking all the cells in the current selection before turning on Data Entry mode. This is useful if you want a user to view a range of cells but not change it or move the active cell. Similarly, if you unlock certain cells, you can restrict the user's movement to the Data Entry area only.
- To prevent someone from activating another workbook, which would turn off Data Entry mode, use the ON.WINDOW function or an Auto_Deactivate macro.

Related Functions

DISABLE.INPUT	Blocks all input to Microsoft Excel
FORMULA	Enters values into a cell or range or onto a chart

ERROR

Specifies what action to take if an error is encountered while a macro is running. Use ERROR to control whether Microsoft Excel error messages are displayed, or to run your own macro when an error is encountered.

Syntax

ERROR(**enable_logical**, macro_ref)

Enable_logical   is a logical value or number that selects or clears error-checking.

- If enable_logical is FALSE or 0, all error-checking is cleared. If error-checking is cleared and an error is encountered while a macro is running, Microsoft Excel ignores it and continues. Error-checking is selected again by an ERROR(TRUE) statement, or when the macro stops running.
- If enable_logical is TRUE or 1, you can either select normal error-checking (by omitting the other argument) or specify a macro to run when an error is encountered by using the macro_ref argument. When normal error-checking is active, the Macro Error dialog box is displayed when an error is encountered. You can halt the macro, start single-stepping through the macro, continue running the macro normally, or go to the macro cell where the error occurred.
- If enable_logical is 2 and macro_ref is omitted, error-checking is normal except that if the user clicks the Cancel button in an alert message, ERROR returns FALSE and the macro is not interrupted.
- If enable_logical is 2 and macro_ref is given, the macro goes to that macro_ref when an error is encountered. If the user clicks the Cancel button in an alert message, FALSE is returned and the macro is not interrupted.

Macro_ref   specifies a macro to run if enable_logical is TRUE, 1, or 2 and an error is encountered. It can be either the name of the macro or a cell reference. If enable_logical is FALSE or 0, macro_ref is ignored.

Important   Both ERROR(FALSE) and ERROR(TRUE, macro_ref) keep Microsoft Excel from displaying any messages at all, including the message asking whether to save changes when you close an unsaved workbook. If you want alert messages but not error messages to be displayed, use ERROR(2, macro_ref).

Remarks

You can use GET.WORKSPACE to determine whether error-checking is on or off.

Examples

`ERROR(FALSE)` clears error-checking.

`ERROR(TRUE, Recover)` selects error-checking and runs the macro named Recover when an error is encountered.

The following macro runs the macro ForceMenus if an error occurs in the current macro:

```
=ERROR(TRUE, ForceMenus)
```

Related Functions

CANCEL.KEY   Disables macro interruption

LAST.ERROR   Returns the reference of the cell where the last error occurred

ON.KEY   Runs a macro when a specified key is pressed

ERRORBAR.X, ERRORBAR.Y

Adds error bars to the selected series in a chart. ERRORBAR.X adds bars showing the error factor for the X (category) axis and works for XY (scatter) charts only. ERRORBAR.Y adds bars showing the error factor for the Y (value) axis for all charts.

Syntax

ERRORBAR.X(include, type, amount, minus)

ERRORBAR.Y(include, type, amount, minus)

Include specifies the type of error value to include:

Include	Type of error value
1 or omitted	Plus and minus
2	Plus
3	Minus
4	None

Type specifies the type of error bars to display:

Type	Type of error displayed
1 or omitted	Fixed amount
2	Percent
3	Multiplying factor standard deviation (default value is 1)
4	Standard error
5	Custom

Amount   is the range of error values to display. This argument depends on the value of type:

If type is	then amount
1 or omitted	Can be any number greater than 0
2	Can be any number greater than 0
3	Can be any number greater than or 0
4	Not required
5	Is the positive amount for custom error bars

Minus   is the negative amount for custom error bars. Applicable only if type is 5.

Remarks

For the amount argument, standard deviation(s) can be calculated using this equation:

$$S.D. = \sqrt{\frac{\sum_{s=1}^{m}\sum_{i=1}^{n} y_{is}^{2}}{(n_y - 1)}}$$

$$M = \frac{\sum_{s=1}^{m}\sum_{i=1}^{n} y_{is}}{n_y}$$

The standard deviation is multiplied by the value specified by amount and the error bars are placed this distance from the arithmetic mean. Therefore, these error bars are plotted along the arithmetic mean, not attached to data series.

Microsoft Excel calculates the standard error using the following equation:

$$S.E. = \sqrt{\frac{\sum_{s=1}^{m}\sum_{i=1}^{n} y_{is}^{2}}{(n_y - 1)(n_y)}}$$

Both the standard deviation and standard error functions use the following variables:

Variable	Equals
s	Series number
i	Point number in series s
m	Number of series for point y in chart
n	Number of points in each series
Yi	Data value of series s and the ith point
Ny	Total number of data values in all series
M	Arithmetic mean

EVALUATE

Evaluates a formula or expression that is in the form of text and returns the result. To run a macro or subroutine, use the RUN function.

Syntax

EVALUATE(**formula_text**)

Formula_text is the expression in the form of text that you want to evaluate.

Remarks

Using EVALUATE is similar to selecting an expression within a formula in the formula bar and pressing the Recalculate key (F9 in Microsoft Excel for Windows and COMMAND+= in Microsoft Excel for the Macintosh). EVALUATE replaces an expression with a value.

Example

Suppose you want to know the value of a cell named LabResult1, LabResult2, or LabResult3, where the 1, 2, or 3 is specified by the name TrialNum whose value may change as the macro runs. You can use the following formula to calculate the value:

```
EVALUATE("LabResult"&TrialNum)
```

Related Function

RUN Runs a macro

EXEC

Starts a separate program. Use EXEC to start other programs with which you want to communicate. Use EXEC with Microsoft Excel's other DDE functions (INITIATE, EXECUTE, and SEND.KEYS) to create a channel to another program and to send keystrokes and commands to the program. (SEND.KEYS is available only in Microsoft Excel for Windows.)

Syntax 1 is for Microsoft Excel for Windows. Syntax 2 is for Microsoft Excel for the Macintosh.

Syntax 1

For Microsoft Excel for Windows

EXEC(program_text, window_num)

Syntax 2

For Microsoft Excel for the Macintosh

EXEC(program_text, , background, preferred_size_only)

Important Microsoft Excel for the Macintosh requires system software version 7.0 or later for the last two arguments of this function.

Program_text is the name, as a text string, of any executable file or, in Microsoft Excel for Windows, any data file that is associated with an executable file.

- Use paths when the file or program to be started is not in the current directory or folder.
- In Microsoft Excel for Windows, program_text can include any arguments and switches that are accepted by the program to be started. Also, if program_text is the name of a file associated with a specific installed program, EXEC starts the program and loads the specified file.

Note In Microsoft Excel for the Macintosh, you must use an extra comma after the program_text argument. This skips the window_num argument that does not apply to the Macintosh.

Window_num is a number from 1 to 3 that specifies how the window containing the program should appear. Window_num is only available for use with Microsoft Excel for Windows. The window_num argument is allowed on the Macintosh, but it is ignored.

Window_num	Window appears
1	Normal size
2 or omitted	Minimized size
3	Maximized size

Background is a logical value that determines whether the program specified by program_text is opened as the active program or in the background, leaving Microsoft

Excel as the active program. If background is TRUE, the program is started in the background; if FALSE or omitted, the program is started in the foreground. Background is only available for use with Microsoft Excel for the Macintosh and system software version 7.0 or later.

Preferred_size_only is a logical value that determines the amount of memory allocated to the program. If preferred_size_only is TRUE, the program is opened with its preferred memory allocation; if FALSE or omitted, it opens with the available memory if greater than its minimum requirement. Preferred_size_only is only available for use with Microsoft Excel for the Macintosh and system software version 7.0 or later. For information about changing the preferred memory size, see your Macintosh documentation.

Remarks

In Microsoft Excel for Windows and in Microsoft Excel for the Macintosh with system software version 7.0, if the EXEC function is successful, it returns the task ID number of the started program. The task ID number is a unique number that identifies a program. Use the task ID number in other macro functions, such as APP.ACTIVATE, to refer to the program. In Microsoft Excel for the Macintosh with system software version 6.0, if EXEC is successful, it returns TRUE. If EXEC is unsuccessful, it returns the #VALUE! error value.

Examples

In Microsoft Excel for Windows, the following macro formula starts the program SEARCH.EXE. Use paths when the file or program to be started is not in the current directory:

```
EXEC("C:\WINDOWS\SEARCH.EXE")
```

The following macro formula starts Microsoft Word for Windows and loads the document SALES.DOC:

```
EXEC("C:\WINWORD\WINWORD.EXE C:\MYFILES\SALES.DOC")
```

In Microsoft Excel for the Macintosh, the following macro formula starts Microsoft Word:

```
EXEC("HARD DISK:APPS:WORD")
```

Related Functions

APP.ACTIVATE Switches to another application

EXECUTE Carries out a command in another application

INITIATE Opens a channel to another application

SEND.KEYS Sends a key sequence to an application

TERMINATE Closes a channel to another application

REQUEST Requests an array of a specific type of information from an application with which you have a dynamic data exchange (DDE) link

POKE Sends data to another application with which you have a dynamic data exchange (DDE) link

EXECUTE

Carries out commands in another program with which you have a dynamic data exchange (DDE) link. Use with EXEC, INITIATE, and SEND.KEYS to run another program through Microsoft Excel. (SEND.KEYS is available only in Microsoft Excel for Windows.)

Important Microsoft Excel for the Macintosh requires system software version 7.0 or later for this function.

Syntax

EXECUTE(**channel_num, execute_text**)

Channel_num is a number returned by a previously run INITIATE function. Channel_num refers to a channel through which Microsoft Excel communicates with another program.

Execute_text is a text string representing commands you want to carry out in the program specified by channel_num. The form of execute_text depends on the program you are referring to. To include specific key sequences in execute_text, use the format described under key_text in the ON.KEY function.

If EXECUTE is not successful, it returns one of the following error values:

Value returned	Situation
#VALUE!	Channel_num is not a valid channel number.
#N/A	The program you are accessing is busy.
#DIV/0!	The program you are accessing does not respond after a certain length of time or you have pressed ESC to cancel.
#REF!	The keys specified in execute_text are refused by the application which you want to access.

Remarks

Commands sent to another program with EXECUTE will not work when a dialog box is displayed in the program. In Microsoft Excel for Windows, you can use SEND.KEYS to send commands that make selections in a dialog box.

Examples

The following macro formula sends the number 25 and a carriage return to the application identified by channel_num 14:

```
EXECUTE(14, "25~")
```

Related Functions

EXEC Starts another application

INITIATE Opens a channel to another application

POKE Sends data to another application

REQUEST Returns data from another application

SEND.KEYS Sends a key sequence to an application

TERMINATE Closes a channel to another application

EXPON

Predicts a value based on the forecast for the prior period, adjusted for the error in that prior forecast.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

EXPON(**inprng**, outrng, damp, stderrs, chart)

Inprng is the input range.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Damp is the damping factor. If omitted, damp is 0.3.

Stderrs is a logical value. If TRUE, standard error values are included in the output table. If FALSE, standard errors are not included.

Chart is a logical value. If TRUE, EXPON generates a chart for the actual and forecast values. If FALSE, the chart is not generated.

Related Function

MOVEAVG Returns values along a moving average trend

EXTEND.POLYGON

Adds vertices to a polygon. This function must immediately follow a CREATE.OBJECT function or another EXTEND.POLYGON function. Use multiple EXTEND.POLYGON functions to create arbitrarily complex polygons. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax

EXTEND.POLYGON(**array**)

Array is an array of values, or a reference to a range of cells containing values, that indicate the position of vertices in the polygon. The position is measured in points and is relative to the upper-left corner of the polygon's bounding rectangle.

- A vertex is a point. Each vertex is defined by a pair of coordinates in one row of array.

- The polygon is defined by the array argument to the CREATE.OBJECT function and to all the immediately following EXTEND.POLYGON functions.
- If the polygon contains many vertices, one array may not be sufficient to define it. If the number of elements in the formula exceeds 1024, you must include additional EXTEND.POLYGON functions. If you're recording a macro, Microsoft Excel automatically records additional EXTEND.POLYGON functions as needed.

Related Functions

CREATE.OBJECT   Creates an object

FORMAT.SHAPE   Inserts, moves, or deletes vertices of the selected polygon

EXTRACT

Equivalent to choosing the Extract command from the Data menu in Microsoft Excel version 4.0. Finds database records that match the criteria defined in the criteria range and copies them into a separate extract range.

Syntax

EXTRACT(unique)

EXTRACT?(unique)

Unique   is a logical value corresponding to the Unique Records Only check box in the Extract dialog box.

- If unique is TRUE, Microsoft Excel selects the check box and excludes duplicate records from the extract list.
- If unique is FALSE or omitted, Microsoft Excel clears the check box and extracts all records matching the criteria.

Related Functions

DATA.FIND   Finds records in a database

SET.CRITERIA   Defines the name Criteria for the selected range on the active sheet

SET.DATABASE   Defines the name Database for the selected range on the active sheet

SET.EXTRACT   Defines the name Extract for the selected range on the active sheet

FCLOSE

Closes the specified file.

Syntax

FCLOSE(**file_num**)

File_num   is the number of the file you want to close. File_num is returned by the FOPEN function that originally opened the file. If file_num is not a valid file number, FCLOSE halts the macro and returns the #VALUE! error value.

Examples

The following function closes the file identified by FileNumber:

```
FCLOSE(FileNumber)
```

Related Functions

CLOSE   Closes the active window

FILE.CLOSE   Closes the active workbook

FOPEN   Opens a file with the type of permission specified

FILE.CLOSE

Equivalent to clicking the Close command on the File menu. Closes the active workbook.

Syntax

FILE.CLOSE(save_logical, route_logical)

Save_logical   is a logical value specifying whether to save the file before closing it.

Save_logical	**Result**
TRUE	Saves the workbook
FALSE	Does not save the workbook
Omitted	If you've made changes to the workbook, displays a dialog box asking if you want to save the workbook

Route_logical   is a logical value that specifies whether to route the file after closing it. This argument is ignored if there is not a routing slip present.

Route_logical	**Result**
TRUE	Routes the file
FALSE	Does not route the file

Omitted   If you've specified recipients for routing, displays a dialog box asking if you want to save the file

Remarks

If you make any changes to the structure of a workbook, such as the name of sheets, their order, and so on, then a message will be displayed reminding you that there are unsaved changes, regardless of the save_logical value.

Note When you use the FILE.CLOSE function, Microsoft Excel does not run any Auto_Close macros before closing the workbook.

Related Functions

CLOSE Closes the active window

CLOSE.ALL Closes all unprotected windows

FCLOSE Closes a text file

FILE.DELETE

Deletes a file from the disk. Although you will normally delete files manually, you can, for example, use FILE.DELETE in a macro to delete temporary files created by the macro.

Syntax

FILE.DELETE(**file_text**)

FILE.DELETE?(file_text)

File_text is the name of the file to delete.

Remarks

- If Microsoft Excel can't find file_text, it displays a message saying that it cannot delete the file. To avoid this, include the entire path in file_text. See the following second and fifth examples. You can also use FILES to generate an array of filenames and then check if the file you want to delete is in the array.
- If a file is open when you delete it, the file is removed from the disk but remains open in Microsoft Excel.
- In the dialog-box form, FILE.DELETE?, you can use an asterisk (*) to represent any series of characters and a question mark (?) to represent any single character. See the following third and sixth examples.

Examples

In Microsoft Excel for Windows, the following macro formula deletes a file called CHART1.XLS from the current directory:

```
FILE.DELETE("CHART1.XLS")
```

The following macro formula deletes a file called 92INFO.XLS kept in the EXCEL\SALES subdirectory:

```
FILE.DELETE("C:\EXCEL\SALES\92INFO.XLS")
```

The following macro formula displays the Delete dialog box listing all documents whose extensions begin with the letters "XL":

```
FILE.DELETE?("*.XL?")
```

In Microsoft Excel for the Macintosh, the following macro formula deletes a file called CHART1 from the current folder:

```
FILE.DELETE("CHART1")
```

The following macro formula deletes a file called 1992 INFO kept in a series of nested folders:

```
FILE.DELETE("HARD DISK:EXCEL 5:SALES WORKSHEETS:1992 INFO")
```

The following macro formula displays the Delete dialog box listing all documents beginning with the word "Clients":

```
FILE.DELETE?("Clients*")
```

Related Functions

FILE.CLOSE	Closes the active workbook
FILES	Returns the filenames in the specified directory or folder

FILES

Returns a horizontal text array of the names of all files in the specified directory or folder. Use FILES to build a list of filenames upon which you want your macro to operate.

Syntax

FILES(directory_text)

Directory_text   specifies which directories or folders to return filenames from.

- Directory_text accepts an asterisk (*) to represent a series of characters and a question mark (?) to represent a single character in filenames.
- If directory_text is not specified, FILES returns filenames from the current directory.

Remarks

If you enter FILES in a single cell, only one filename is returned. You will normally use FILES with SET.NAME to assign the returned array to a name. See the last example below.

Tips   You can use COLUMNS to count the number of entries in the returned array. You can use TRANSPOSE to change a horizontal array to a vertical one.

Examples

In Microsoft Excel for Windows, the following macro formula returns the names of all files starting with the letter F in the current directory or folder:

```
FILES("F*.*")
```

When entered as an array formula in several cells, the following macro formula returns the filenames in the current directory to those cells. If the directory contains fewer files than can fit in the selected cells, the #N/A error value appears in the extra cells.

```
FILES()
```

In Microsoft Excel for Windows, the following macro formula returns all files starting with "SALE" and ending with the .XLS extension in the \EXCEL\CHARTS subdirectory:

```
FILES("C:\EXCEL\CHARTS\SALE*.XLS")
```

In Microsoft Excel for the Macintosh, the following macro formula returns all files starting with "SALE" in the nested CHART folder:

```
FILES("DISK:EXCEL:CHART:SALE*")
```

The following macro stores the names of the files in the current directory in the named array FileArray

```
SET.NAME("FileArray",FILES())
```

Related Functions

DOCUMENTS	Returns the names of the specified open workbooks
FILE.DELETE	Deletes a file
OPEN	Opens a workbook
SET.NAME	Defines a name as a value

FILL.AUTO

Equivalent to copying cells or automatically filling a selection by dragging the fill selection handle with the mouse (the AutoFill feature).

Syntax

FILL.AUTO(destination_ref, copy_only)

Destination_ref   is the range of cells into which you want to fill data. The top, bottom, left, or right end of destination_ref must include all of the cells in the source reference (the current selection).

Copy_only   is a number specifying whether to copy cells or perform an AutoFill operation.

Value	Result
0 or omitted	Normal AutoFill
1 or TRUE	Copy cells

2	Copy formats
3	Fill values
4	Increment
5	Increment by day
6	Increment by weekday
7	Increment by month
8	Increment by year
9	Linear trend
10	Growth trend

Related Functions

COPY	Copies and pastes data or objects
DATA.SERIES	Fills a range of cells with a series of numbers or dates

FILL.DOWN, FILL.LEFT, FILL.RIGHT, FILL.UP

Equivalent to clicking the Down, Left, Right, and Up commands, respectively, on the Fill submenu of the Edit menu.

Syntax

FILL.DOWN( )

FILL.LEFT( )

FILL.RIGHT( )

FILL.UP( )

FILL.DOWN copies the contents and formats of the cells in the top row of a selection into the rest of the rows in the selection.

FILL.LEFT copies the contents and formats of the cells in the right column of a selection into the rest of the columns in the selection.

FILL.RIGHT copies the contents and formats of the cells in the left column of a selection into the rest of the columns in the selection.

FILL.UP copies the contents and formats of the cells in the bottom row of a selection into the rest of the rows in the selection.

Remarks

If you have a multiple selection, each range in the selection is filled separately with the contents of the source range.

Related Functions

COPY   Copies and pastes data or objects

DATA.SERIES   Fills a range of cells with a series of numbers or dates

FILL.AUTO   Copies cells or automatically fills a selection

FORMULA.FILL   Enters a formula in the specified range

FILL.GROUP

Equivalent to choosing the Across Worksheets command from the Fill submenu on the Edit menu. Copies the contents of the active worksheet's selection to the same area on all other worksheets in the group. Use FILL.GROUP to fill a range of cells on all worksheets in a group at once.

Syntax

FILL.GROUP(**type_num**)

FILL.GROUP?(type_num)

Type_num   is a number from 1 to 3 that corresponds to the choices in the Fill Across Worksheets dialog box.

Type_num	**Type of information filled**
1	All
2	Contents
3	Formats

Related Functions

NEW   Creates a new workbook

WORKBOOK.SELECT   Selects one or more sheets in a workbook

FILTER

Filters lists of data one column at a time. Only one list can be filtered on any one sheet at a time.

Syntax

FILTER(field_num, criteria1, operation, criteria2)

FILTER?(field_num, criteria1, operation, criteria2)

Field_num is the number of the field that you want to filter. Fields are numbered from left to right starting with 1.

Criteria1 is a text string specifying criteria for filtering a list, such as ">2". If you want to include all items in the list, omit this argument.

Operation is a number that specifies how you want criteria2 used with criteria1:

Number	Operation Used
1	AND
2	OR

Criteria2 is a text string specifying criteria for filtering a list, such as ">2". If you include this argument, operation is required.

Remarks

If you omit all arguments, FILTER toggles the display of filter arrows.

Related Function

FILTER.ADVANCED Lets you set options for filtering a list

FILTER.ADVANCED

Equivalent to choosing the Advanced Filter command from the Filter submenu on the Data menu. Lets you set options for filtering a list.

Syntax

FILTER.ADVANCED(**operation**, **list_ref**, criteria_ref, copy_ref, unique)

FILTER.ADVANCED?(operation, list_ref, criteria_ref, copy_ref, unique)

Operation is a number specifying whether to copy the filter list to a new location. To filter a list without copying, use 1; to copy the filter list to a new location, use 2.

List_ref specifies the location of the list to be filtered. If operation is 1, then list_ref must be on the active sheet.

Criteria_ref is a reference to a range containing criteria for filtering the list. If omitted, uses "All" as the criteria.

Copy_ref is a reference on the active sheet where you want the filtered list copied. Ignored if operation is 1.

Unique is a logical value that specifies whether only unique records are displayed. To display only unique records, use TRUE. To display all records that match the criteria, use FALSE or omit this argument.

Related Function

FILTER	Filters lists of data one column at a time

FILTER.SHOW.ALL

Equivalent to choosing the Show All command from the Filter submenu on the Data menu. Displays all items in a filtered list.

Syntax

FILTER.SHOW.ALL()

FIND.FILE

Equivalent to choosing the Find File command from the File menu in Microsoft Excel version 5.0. Lets you search for files based on criteria such as author or creation date.

Syntax

FIND.FILE?( )

Remarks

This function has a dialog-box form only.

FONT

Equivalent to clicking the Font command on the Options menu in Microsoft Excel for the Macintosh version 1.5 or earlier. This function is included only for macro compatibility. Sets the font for the Normal style. Microsoft Excel now uses the FONT.PROPERTIES and DEFINE.STYLE functions. For more information, see FONT.PROPERTIES and DEFINE.STYLE.

Syntax

FONT(**name_text, size_num**)

FONT?(name_text, size_num)

Related Functions

DEFINE.STYLE	Creates or changes a cell style
FONT.PROPERTIES	Sets various font properties

FONT.PROPERTIES

Equivalent to choosing the Cells command from the Format menu. Applies a font and other attributes to the selection. Applies to cells, charts, and text boxes and buttons on worksheets and macro sheets.

Syntax

FONT.PROPERTIES(font, font_style, size, strikethrough, superscript, subscript, outline, shadow, underline, color, normal, background, start_char, char_count)

FONT.PROPERTIES?(font, font_style, size, strikethrough, superscript, subscript, outline, shadow, underline, color, normal, background, start_char, char_count)

Arguments correspond to check boxes or options in the Font tab on the Format Cells dialog box. Arguments that correspond to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the format is not changed.

Font   is the name of the font as it appears on the Font tab. For example, Courier is a font name.

Font_style   is the name of the font style as it appears on the Font tab. For example, Bold Italic is a font style.

Size   is the font size, in points.

Strikethrough   corresponds to the Strikethrough check box.

Superscript   corresponds to the Superscript check box

Subscript   corresponds to the Subscript check box

Outline   corresponds to the Outline check box. Outline fonts are available in Microsoft Excel for the Macintosh. For macro compatibility, this argument is ignored by Microsoft Excel for Windows..

Shadow   corresponds to the Shadow check box. Shadow fonts are available in Microsoft Excel for the Macintosh. For macro compatibility, this argument is ignored by Microsoft Excel for Windows.

Note   For macro compatibility with Microsoft Excel for the Macintosh, the presence of the outline and shadow arguments do not prevent the macro from working on Microsoft Excel for Windows, nor does their absence prevent it from working on the Macintosh.

Underline   corresponds to the Underline Drop-down box.

Underline	Type applied
0	None
1	Single
2	Double
3	Single Accounting

4	Double Accounting

Color is a number from 0 to 56 corresponding to the colors listed in the Color box; 0 corresponds to automatic color.

Normal corresponds to the Normal Font check box. Applies the default font for your system

Background is a number from 1 to 3 specifying which type of background to apply to text in a chart.

Background	Type of background applied
1	Automatic
2	Transparent
3	Opaque

Start_char specifies the first character to be formatted. If start_char is omitted, it is assumed to be 1 (the first character in the cell or text box).

Char_count specifies how many characters to format. If char_count is omitted, Microsoft Excel formats all characters in the cell or text box starting at start_char.

Remarks

Some extended TrueType styles do not have corresponding arguments to FONT.PROPERTIES. To access an extended TrueType font style, append the style name to the font name in the font argument. For example, the font Taipei can be formatted in an upside-down style by specifying "Taipei Upside-down" as the font argument. For more information about TrueType, see your Microsoft Windows documentation.

Related Functions

ALIGNMENT Aligns or wraps text in cells

FORMAT.NUMBER Applies a number format to the selection

FORMAT.TEXT Formats a worksheet text box or a chart text item

FOPEN

Opens a file with the type of permission specified. Unlike OPEN, FOPEN does not load the file into memory and display it; instead, FOPEN establishes a channel with the file so that you can exchange information with it. If the file is opened successfully, FOPEN returns a file ID number. If it can't open the file, FOPEN returns the #N/A error value. Use the file ID number with other file functions (such as FREAD, FWRITE, and FSIZE) when you want to get information from or send information to the file.

Syntax

FOPEN(**file_text**, access_num)

File_text   is the name of the file you want to open.

Access_num   is a number from 1 to 3 specifying what type of permission to allow to the file:

Access_num	**Type of permission**
1 or omitted	Can read and write to the file (read/write permission)
2	Can read the file, but can't write to the file (read-only permission)
3	Creates a new file with read/write permission

- If the file doesn't exist and access_num is 3, FOPEN creates a new file.
- If the file does exist and access_num is 3, FOPEN replaces the contents of the file with any information you supply using the FWRITE or FWRITELN functions.
- If the file doesn't exist and access_num is 1 or 2, FOPEN returns the #N/A error value.

Remarks

Use FCLOSE to close a file after you finish using it.

Example

The following function opens a file identified as FileName using read-only permission:

```
FOPEN(FileName, 2)
```

Related Functions

FCLOSE   Closes a text file

FREAD   Reads characters from a text file

FWRITE   Writes characters to a text file

OPEN   Opens a workbook

FOR

Starts a FOR-NEXT loop. The instructions between FOR and NEXT are repeated until the loop counter reaches a specified value. Use FOR when you need to repeat instructions a specified number of times. Use FOR.CELL when you need to repeat instructions over a range of cells.

Syntax

FOR(**counter_text, start_num, end_num**, step_num)

Counter_text is the name of the loop counter in the form of text.

Start_num is the value initially assigned to counter_text.

End_num is the last value assigned to counter_text.

Step_num is a value added to the loop counter after each iteration. If step_num is omitted, it is assumed to be 1.

Remarks

- Microsoft Excel follows these steps as it executes a FOR-NEXT loop:

Step	Action
1	Sets counter_text to the value start_num.
2	If counter_text is greater than end_num (or less than end_num if step_num is negative), the loop ends, and the macro continues with the function after the NEXT function. If counter_text is less than or equal to end_num (or greater than or equal to end_num if step_num is negative), the macro continues in the loop.
3	Carries out functions up to the following NEXT function. The NEXT function must be below the FOR function and in the same column.
4	Adds step_num to the loop counter.
5	Returns to the FOR function and proceeds as described in step 2.

- You can interrupt a FOR-NEXT loop by using the BREAK function.

Example

The following macro starts a FOR-NEXT loop that is executed once for every open window:

```
FOR("Counter", 1, COLUMNS(WINDOWS()))
```

Related Functions

BREAK   Interrupts a FOR-NEXT, FOR.CELL-NEXT, or WHILE-NEXT loop

FOR.CELL   Starts a FOR.CELL-NEXT loop

NEXT   Ends a FOR-NEXT, FOR.CELL-NEXT, or WHILE-NEXT loop

WHILE   Starts a WHILE-NEXT loop

FOR.CELL

Starts a FOR.CELL-NEXT loop. This function is similar to FOR, except that the instructions between FOR.CELL and NEXT are repeated over a range of cells, one cell at a time, and there is no loop counter.

Syntax

FOR.CELL(**ref_name**, area_ref, skip_blanks)

Ref_name   is the name in the form of text that Microsoft Excel gives to the one cell in the range that is currently being operated on; ref_name refers to a new cell during each loop.

Area_ref   is the range of cells on which you want the FOR.CELL-NEXT loop to operate and can be a multiple selection. If area_ref is omitted, it is assumed to be the current selection.

Skip_blanks   is a logical value specifying whether Microsoft Excel skips blank cells as it operates on the cells in area_ref.

Skip_blanks	**Result**
TRUE	Skips blank cells in area_ref
FALSE or omitted	Operates on all cells in area_ref

Remarks

FOR.CELL operates on each cell in a row from left to right one area at a time before moving to the next row in the selection.

Example

The following macro starts a FOR.CELL-NEXT loop and uses the name CurrentCell to refer to the cell in the range that is currently being operated on:

```
FOR.CELL("CurrentCell", SELECTION(), TRUE)
```

Related Functions

BREAK   Interrupts a FOR-NEXT, FOR.CELL-NEXT, or WHILE-NEXT loop

FOR   Starts a FOR-NEXT loop

NEXT   Ends a FOR-NEXT, FOR.CELL-NEXT, or WHILE-NEXT loop

WHILE Starts a WHILE-NEXT loop

FORMAT.AUTO

Equivalent to clicking the AutoFormat command on the Format menu when a worksheet is active or clicking the AutoFormat button. Formats the selected range of cells from a built-in gallery of formats.

Syntax

FORMAT.AUTO(format_num, number, font, alignment, border, pattern, width)

FORMAT.AUTO?(format_num, number, font, alignment, border, pattern, width)

Format_num is a number from 1 to 17 corresponding to the formats in the Table Format list box in the AutoFormat dialog box.

Format_num	**Table Format**
0	None
1 or omitted	Classic 1
2	Classic 2
3	Classic 3
4	Accounting 1
5	Accounting 2
6	Accounting 3
7	Colorful 1
8	Colorful 2
9	Colorful 3
10	List 1

11	List 2
12	List 3
13	3D Effects 1
14	3D Effects 2
15	Japan 1 (Far East versions of Microsoft Excel only)
16	Japan 2 (Far East versions of Microsoft Excel only)
17	Accounting 4
18	Simple

The following arguments are logical values corresponding to the Formats To Apply check boxes in the AutoFormat dialog box. If an argument is TRUE or omitted, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box.

Number   corresponds to the Number check box.

Font   corresponds to the Font check box.

Alignment   corresponds to the Alignment check box.

Border   corresponds to the Border check box.

Pattern   corresponds to the Pattern check box.

Width   corresponds to the Column Width/Row Height check box.

Related Functions

ALIGNMENT   Aligns or wraps text in cells

BORDER   Adds a border to the selected cell or object

FONT.PROPERTIES   Applies a font to the selection

FORMAT.NUMBER   Applies a number format to the selection

PATTERNS   Changes the appearance of the selected object

FORMAT.CHART

Equivalent to choosing the Options button in the Chart Type dialog box, which is available when you choose the Chart Type command from the Format menu when a chart is active. Formats the chart according to the arguments you specify.

Syntax

FORMAT.CHART(layer_num, view, overlap, angle, gap_width, gap_depth, chart_depth, doughnut_size, axis_num, drop, hilo, up_down, series_line, labels, vary)

FORMAT.CHART?(layer_num, view, overlap, angle, gap_width, gap_depth, chart_depth, doughnut_size, axis_num, drop, hilo, up_down, series_line, labels, vary)

Several of the following arguments are logical values corresponding to check boxes in the Options tab of Format (chart type) Group dialog box. If an argument is TRUE, Microsoft Excel selects the corresponding check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the setting is unchanged.

Layer_num   is a number specifying which chart you want to change.

View   is a number specifying one of the subtypes in the Subtype tab of the Format (type) Group dialog box. The subtype varies depending on the type of chart.

Overlap   is a number from -100 to 100 specifying how you want bars or columns to be positioned. It corresponds to the Overlap edit box in the Options tab on the Format Bar Group Dialog box, which appears when you choose the Bar Group from the Format menu. Overlap is ignored if type_num is not 2 or 3 (bar or column chart).

- If overlap is positive, it specifies the percentage of overlap you want for bars or columns. For example, 50 would cause one-half of a bar or column to be covered by an adjacent bar or column. A value of zero prevents bars or columns from overlapping.
- If overlap is negative, then bars or columns are separated by the specified percentage of the maximum available distance between any two bars or columns.
- If overlap is omitted, it is assumed to be 0 (bars or columns do not overlap), or it is unchanged if a value was previously set.

Angle   is a number from 0 to 360 specifying the angle of the first pie or doughnut slice (in degrees) if the chart is a pie or doughnut chart. If angle is omitted, it is assumed to be 0, or it is unchanged if a value was previously set.

Gap_width   is a number from 0 to 500 specifying the space between bar or column clusters as a percentage of the width of a bar or column. It corresponds to the Gap Width edit box in the Options tab on the Format Bar Group Dialog box, which appears when you choose the Bar Group from the Format menu.

- Gap_width is ignored if type_num is not 2, 3, 8, or 12 (bar or column chart).
- If Gap_width is omitted, it is assumed to be 50, or it is unchanged if a value was previously set.

The next two arguments are for 3-D charts only, and correspond to check boxes in the Options tab of Format (chart type) Group dialog box.

Gap_depth is a number from 0 to 500 specifying the depth of the gap in front of and behind a bar, column, area, or line as a percentage of the depth of the bar, column, area, or line.

- Gap_depth is ignored if the chart is a pie chart or if it is not a 3-D chart.
- If gap_depth is omitted and the chart is a 3-D chart, gap_depth is assumed to be 50, or it is unchanged if a value was previously set. If gap_depth is omitted and the view is side-by-side, stacked, or stacked 100%, gap_depth is assumed to be 0, or it is unchanged if a value was previously set.

Chart_depth is a number from 20 to 2000 specifying the visual depth of the chart as a percentage of the width of the chart.

- Chart_depth is ignored if the chart is not a 3-D chart.
- If Chart_depth is omitted, it is assumed to be 100, or it is unchanged if a value was previously set.

Doughnut_size specifies the size of the hole in a doughnut chart. Can be a value from 10% - 90%. Default is 50%.

Axis_num is a number specifying whether to plot the chart on the primary axis or the secondary axis.

Drop corresponds to the Drop Lines check box. Drop is available only for area and line charts.

Hilo corresponds to the Hi-Lo Lines check box. Hilo is available only for 2-D line charts.

The next four arguments are logical values corresponding to check boxes in the Options tab of the Format (chart type) Group dialog box. If an argument is TRUE, Microsoft Excel selects the corresponding check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the setting is unchanged.

Up_down corresponds to the Up/Down Bars check box. Up_down is available only for 2-D line charts.

Series_line corresponds to the Series Lines check box. Series_line is available only for 2-D stacked bar and column charts.

Labels corresponds to the Radar Axis Labels check box. Labels is available only for radar charts.

Vary corresponds to the Vary Colors By Point check box. Vary applies only to charts with one data series and is not available for area charts.

Related Functions

FORMAT.MAIN Formats a chart according to the arguments you specify

FORMAT.OVERLAY Formats an overlay chart

FORMAT.CHARTTYPE

Changes the chart type for a selected data series, a group of data series, or an entire chart.

Syntax

FORMAT.CHARTTYPE(**apply_to**, group_num, dimension, **type_num**)

FORMAT.CHARTTYPE?(apply_to, group_num, dimension, type_num)

Apply_to is a number from 1 to 3 specifying what part of a chart the new chart type effects.

Value	Part of chart
1	Selected data series
2	Group of data series
3	Entire chart

Group_num corresponds to the number of the group you want to change as listed in the Group list box of the Chart Type dialog box, which appears when you click Chart Type from the Format menu while a chart is active. Groups are numbered starting with 1 for the group at the top of the list. This argument is required if apply_to equals 2; otherwise it is ignored.

Dimension specifies whether to apply a 2-D or 3-D chart type. Use 1 for a 2-D chart type or 2 for a 3-D chart type. If omitted, uses the same dimension as the series, group, or chart to be changed.

Type_num specifies the chart type to apply. Meaning of type_num varies depending on the value of dimension:

Type_num	Chart type if dimension is 1
1	Area or 3-D area
2	Bar or 3-D bar
3	Column or 3-D column
4	Line or 3-D line
5	Pie or 3-D pie

6	Doughnut or 3-D surface
7	Radar
8	XY (scatter)

Related Function

FORMAT.CHART   Formats the selected chart

FORMAT.FONT

Equivalent to choosing the Cells command from the Format menu, and then selecting Font tab from the Format Cells dialog box. This function is included for compatibility with Microsoft Excel version 4.0. Use FONT.PROPERTIES to set various font properties. FORMAT.FONT has three syntax forms. Syntax 1 is for cells; syntax 2 is for text boxes and buttons; syntax 3 is used with all chart items (axes, labels, text, and so on).

Syntax 1

Cells

FORMAT.FONT(name_text, size_num, bold, italic, underline, strike, color, outline, shadow)

FORMAT.FONT?(name_text, size_num, bold, italic, underline, strike, color, outline, shadow)

Syntax 2

Text boxes and buttons on worksheets and macro sheets

FORMAT.FONT(name_text, size_num, bold, italic, underline, strike, color, outline, shadow, object_id_text, start_num, char_num)

FORMAT.FONT?(name_text, size_num, bold, italic, underline, strike, color, outline, shadow, object_id_text, start_num, char_num)

Syntax 3

Chart items including unattached chart text

FORMAT.FONT(color, backgd, apply, name_text, size_num, bold, italic, underline, strike, outline, shadow, object_id_text, start_num, char_num)

FORMAT.FONT?(color, backgd, apply, name_text, size_num, bold, italic, underline, strike, outline, shadow, object_id_text, start_num, char_num)

Arguments correspond to check boxes and list box items in the Font tab on the Format Cells dialog box. Arguments that correspond to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the format is not changed.

Name_text   is the name of the font as it appears in the Font tab. For example, Courier is a font name.

Size_num   is the font size, in points.

Bold corresponds to the Bold item in the Font Style list box. Makes the selection bold, if applicable.

Italic corresponds to the Italic item in the Font Style list box. Makes the selection italic, if applicable.

Underline corresponds to the Underline check box.

Strike corresponds to the Strikethrough check box.

Color is a number from 0 to 56 corresponding to the colors in the Font tab; 0 corresponds to automatic color.

Outline corresponds to the Outline check box. Outline fonts are available in Microsoft Excel for the Macintosh. For macro compatibility, this argument is ignored by Microsoft Excel for Windows.

Shadow corresponds to the Shadow check box. Shadow fonts are available in Microsoft Excel for the Macintosh. For macro compatibility, this argument is ignored by Microsoft Excel for Windows.

Note For macro compatibility with Microsoft Excel for the Macintosh, the presence of the outline and shadow arguments do not prevent the macro from working on Microsoft Excel for Windows, nor does their absence prevent it from working on the Macintosh.

Object_id_text identifies the text box you want to format (for example, "Text 1", "Text 2", and so on). You can also use the object number alone without the text identifier. For compatibility with earlier versions of Microsoft Excel. This argument is ignored in Microsoft Excel version 5.0 or later.

Start_num specifies the first character to be formatted. If start_num is omitted, it is assumed to be 1 (the first character in the text box).

Char_num specifies how many characters to format. If char_num is omitted, Microsoft Excel formats all characters in the text box starting at start_num.

Backgd is a number from 1 to 3 specifying which type of background to apply to text in a chart.

Backgd	Type of background applied
1	Automatic
2	Transparent
3	Opaque

Apply corresponds to the Apply To All check box. This argument applies to data labels only.

Remarks

Some extended TrueType styles do not have corresponding arguments to FORMAT.FONT. To access an extended TrueType font style, append the style name to the font name in name_text. For example, the font Taipei can be formatted in an upside-down style by specifying "Taipei Upside-down" as the name_text argument. For more information about TrueType, see your Microsoft Windows documentation.

Related Functions

ALIGNMENT   Aligns or wraps text in cells

FONT.PROPERTIES   Sets various font attributes

FORMAT.NUMBER   Applies a number format to the selection

FORMAT.TEXT   Formats a worksheet text box or a chart text item

FORMAT.LEGEND

Equivalent to clicking the Selected Legend command on the Format menu when a chart is active. Determines the position and orientation of the legend on a chart and returns TRUE; returns an error message if the legend is not already selected.

Syntax

FORMAT.LEGEND(**position_num**)

FORMAT.LEGEND?(position_num)

Position_num   is a number from 1 to 5 specifying the position of the legend.

Position_num	Position of legend
1	Bottom
2	Corner
3	Top
4	Right
5	Left

Related Functions

FORMAT.MOVE   Moves the selected object

FORMAT.SIZE   Sizes an object

LEGEND   Adds or deletes a chart legend

FORMAT.MAIN

Equivalent to clicking the Main Chart command on the Format menu in Microsoft Excel version 4.0. Formats a chart according to the arguments you specify. This function is included for compatibility with Microsoft Excel version 4.0. In Microsoft Excel version 5.0 or later, this is equivalent to clicking the Chart Type command on the Format menu. You can also use the FORMAT.CHART function.

Syntax

FORMAT.MAIN(**type_num**, view, overlap, gap_width, vary, drop, hilo, angle, gap_depth, chart_depth, up_down, series_line, labels, doughnut_size)

FORMAT.MAIN?(type_num, view, overlap, gap_width, vary, drop, hilo, angle, gap_depth, chart_depth, up_down, series_line, labels, doughnut_size)

Type_num   is a number specifying the type of chart.

Type_num	**Chart**
1	Area
2	Bar
3	Column
4	Line
5	Pie
6	XY (Scatter)
7	3-D Area
8	3-D Column
9	3-D Line
10	3-D Pie
11	Radar
12	3-D Bar

13	3-D Surface
14	Doughnut

View is a number specifying one of the views in the Data View box in the Main Chart dialog box. The view varies depending on the type of chart.

Overlap is a number from -100 to 100 specifying how you want bars or columns to be positioned. It corresponds to the Overlap box in the Main Chart dialog box. Overlap is ignored if type_num is not 2 or 3 (bar or column chart).

- If overlap is positive, it specifies the percentage of overlap you want for bars or columns. For example, 50 would cause one-half of a bar or column to be covered by an adjacent bar or column. A value of zero prevents bars or columns from overlapping.
- If overlap is negative, then bars or columns are separated by the specified percentage of the maximum available distance between any two bars or columns.
- If overlap is omitted, it is assumed to be 0 (bars or columns do not overlap), or it is unchanged if a value was previously set.

Gap_width is a number from 0 to 500 specifying the space between bar or column clusters as a percentage of the width of a bar or column. It corresponds to the Gap Width box in the Main Chart dialog box.

- Gap_width is ignored if type_num is not 2, 3, 8, or 12 (bar or column chart).
- If gap_width is omitted, it is assumed to be 50, or it is unchanged if a value was previously set.

Several of the following arguments are logical values corresponding to check boxes in the Main Chart dialog box. If an argument is TRUE, Microsoft Excel selects the corresponding check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the setting is unchanged.

Vary corresponds to the Vary By Categories check box. Vary applies only to charts with one data series and is not available for area charts.

Drop corresponds to the Drop Lines check box. Drop is available only for area and line charts.

Hilo corresponds to the Hi-Lo Lines check box. Hilo is available only for line charts.

Angle is a number from 0 to 360 specifying the angle of the first pie slice (in degrees) if the chart is a pie chart. If angle is omitted, it is assumed to be 0, or it is unchanged if a value was previously set.

The next two arguments are for 3-D charts only.

Gap_depth is a number from 0 to 500 specifying the depth of the gap in front of and behind a bar, column, area, or line as a percentage of the depth of the bar, column, area, or line.

- Gap_depth is ignored if the chart is a pie chart or if it is not a 3-D chart.

- If gap_depth is omitted and the chart is a 3-D chart, gap_depth is assumed to be 50, or it is unchanged if a value was previously set. If gap_depth is omitted and the view is side-by-side, stacked, or stacked 100%, gap_depth is assumed to be 0, or it is unchanged if a value was previously set.

Chart_depth   is a number from 20 to 2000 specifying the visual depth of the chart as a percentage of the width of the chart. Chart_depth corresponds to the Chart Depth box in the Main Chart dialog box.

- Chart_depth is ignored if the chart is not a 3-D chart.
- If chart_depth is omitted, it is assumed to be 100, or it is unchanged if a value was previously set.

The next three arguments are logical values corresponding to check boxes in the Main Chart dialog box. If an argument is TRUE, Microsoft Excel selects the corresponding check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the setting is unchanged. The final argument is for compatibility with Microsoft Excel version 4.0.

Up_down   corresponds to the Up/Down Bars check box. Up_down is available only for line charts.

Series_line   corresponds to the Series Lines check box. Series_line is available only for stacked bar and column charts.

Labels   corresponds to the Radar Axis Labels check box. Labels is available only for radar charts.

Doughnut_size   specifies the size of the hole in a doughnut chart. Can be a value from 10% - 90%. Default is 50%

Related Functions

FORMAT.CHART	Formats a chart
FORMAT.OVERLAY	Formats an overlay chart

FORMAT.MOVE

Equivalent to moving an object with the mouse. Moves the selected object to the specified position and, if successful, returns TRUE. If the selected object cannot be moved, FORMAT.MOVE returns FALSE. There are three syntax forms of this function. Use syntax 1 to move worksheet objects. Use syntax 2 to move chart items. Use syntax 3 to move pie-chart and doughnut-chart items. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax 1	Moves worksheet items
Syntax 2	Moves chart items
Syntax 3	Moves pie-chart and doughnut-chart items

FORMAT.MOVE SYNTAX 1

Equivalent to moving an object with the mouse. Moves the selected object to the specified position and, if successful, returns TRUE. If the selected object cannot be moved, FORMAT.MOVE returns FALSE. There are three syntax forms of this function. Use syntax 1 to move worksheet objects. Use syntax 2 to move chart items. Use syntax 3 to move pie-chart and doughnut-chart items. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax

FORMAT.MOVE(**x_offset**, **y_offset**, reference)

FORMAT.MOVE?(x_offset, y_offset, reference)

X_offset specifies the horizontal position to which you want to move the object and is measured in points from the upper-left corner of the object to the upper-left corner of the cell specified by reference. A point is 1/72nd of an inch.

Y_offset specifies the vertical position to which you want to move the object and is measured in points from the upper-left corner of the object to the upper-left corner of the cell specified by reference.

Reference specifies which cell or range of cells to place the object in relation to.

- If reference is a range of cells, only the upper-left cell is used.
- If reference is omitted, it is assumed to be cell A1.

Remarks

The position of an object is based on its upper-left corner. For ovals and arcs, the position is based on the upper-left corner of the bounding rectangle of the object.

Example

The following macro formula moves an object on the active worksheet so that it is 10 points horizontally offset and 15 points vertically offset from cell D4:

```
FORMAT.MOVE(10, 15, !$D$4)
```

Related Functions

CREATE.OBJECT Creates an object

FORMAT.SIZE Sizes an object

WINDOW.MOVE Moves a window

Syntax 2 Moves chart items

Syntax 3 Moves pie-chart and doughnut-chart items

FORMAT.MOVE SYNTAX 2

Equivalent to moving an object with the mouse. Moves the base of the selected object to the specified position and, if successful, returns TRUE. If the selected object cannot be moved, FORMAT.MOVE returns FALSE. There are three syntax forms of this function. Use syntax 3 to move pie-chart and doughnut-chart items. Use syntax 1 to move worksheet objects. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax

FORMAT.MOVE(**x_pos**, **y_pos**)

FORMAT.MOVE?(x_pos, y_pos)

X_pos specifies the horizontal position to which you want to move the object and is measured in points from the base of the object to the lower-left corner of the window. A point is 1/72nd of an inch.

Y_pos specifies the vertical position to which you want to move the object and is measured in points from the base of the object to the lower-left corner of the window.

Remarks

- The base of a text label on a chart is the lower-left corner of the text rectangle.
- The base of an arrow is the end without the arrowhead.
- The base of a pie slice is the point.

Example

On a chart, the following macro formula moves the base of the selected chart object 10 points to the right of and 20 points above the lower-left corner of the window:

```
FORMAT.MOVE(10, 20)
```

Related Functions

FORMAT.SIZE	Sizes an object
WINDOW.MOVE	Moves a window
Syntax 1	Moves worksheet items
Syntax 3	Moves pie-chart and doughnut-chart items

FORMAT.MOVE SYNTAX 3

Equivalent to exploding by moving a pie-chart or doughnut-chart slice with the mouse. Sets the percentage of pie-chart or doughnut-chart slice explosion, and, if successful, returns TRUE. If the selected object cannot be exploded, returns FALSE. There are three syntax forms of this function. Use syntax 1 to move worksheet items. Use syntax 2 to move chart items. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax

FORMAT.MOVE(**explosion_num**)

Explosion_num is a number specifying the explosion percentage for the selected pie slice or the entire chart (if the series is selected). Zero is no explosion (the tip of the slice is in the center of the pie).

Related Functions

FORMAT.SIZE	Sizes an object
Syntax 1	Moves worksheet items
Syntax 2	Moves chart items
WINDOW.MOVE	Moves a window

FORMAT.NUMBER

Equivalent to choosing the Number tab in the Format Cells dialog box, which appears when you choose Cells from the Format menu. Formats numbers, dates, and times in the selected cells, data labels, and axis labels on charts. Use FORMAT.NUMBER to apply built-in formats or to create and apply custom formats.

Syntax

FORMAT.NUMBER(**format_text**)

FORMAT.NUMBER?(format_text)

Format_text is a format string, such as "#, ##0.00", specifying which format to apply to the selection.

Related Functions

DELETE.FORMAT	Deletes the specified custom number format
FONT.PROPERTIES	Applies a font to the selection
FORMAT.TEXT	Formats a sheet text box or a chart text item

FORMAT.OVERLAY

Equivalent to clicking the Overlay command on the Format menu in Microsoft Excel version 4.0. Formats the overlay chart according to the arguments you specify.

Syntax

FORMAT.OVERLAY(**type_num**, view, overlap, gap_width, vary, drop, hilo, angle, series_dist, series_num, up_down, series_line, labels)

FORMAT.OVERLAY?(type_num, view, overlap, gap_width, vary, drop, hilo, angle, series_dist, series_num, up_down, series_line, labels)

Type_num is a number specifying the type of chart.

Type_num	Chart
1	Area
2	Bar
3	Column
4	Line
5	Pie
6	XY (Scatter)
11	Radar
14	Doughnut

View is a number specifying one of the views in the Data View box in the Overlay dialog box. The view varies depending on the type of chart.

Overlap is a number from -100 to 100 specifying how you want bars or columns to be positioned. It corresponds to the Overlap box in the Overlay dialog box. Overlap is ignored if type_num is not 2 or 3 (bar or column chart).

- If overlap is positive, it specifies the percentage of overlap you want for bars or columns. For example, 50 would cause one-half of a bar or column to be covered by an adjacent bar or column.
- If overlap is negative, then bars or columns are separated by the specified percentage of the maximum available distance between any two bars or columns.
- If overlap is omitted, it is assumed to be 0 (bars or columns do not overlap), or it is unchanged if a value was previously set.

Gap_width is a number from 0 to 500 specifying the space between bar or column clusters as a percentage of the width of a bar or column.

- Gap_width is ignored if type_num is not 2 or 3 (bar or column chart).

- If gap_width is omitted, it is assumed to be 50, or it is unchanged if a value was previously set.

Several of the following arguments are logical values corresponding to check boxes in the Overlay dialog box. If an argument is TRUE, Microsoft Excel selects the corresponding check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the setting is unchanged.

Vary corresponds to the Vary By Categories check box. Vary is not available for area charts.

Drop corresponds to the Drop Lines check box. Drop is available only for area and line charts.

Hilo corresponds to the Hi-Lo Lines check box. Hilo is available only for line charts.

Angle is a number from 0 to 360 specifying the angle of the first pie slice (in degrees) if the chart is a pie chart. If angle is omitted, it is assumed to be 0, or it is unchanged if a value was previously set.

Series_dist is the number 1 or 2 and specifies automatic or manual series distribution.

- If series_dist is 1 or omitted, Microsoft Excel uses automatic series distribution.
- If series_dist is 2, Microsoft Excel uses manual series distribution, and you must specify which series is first in the distribution by using the series_num argument.

Series_num is the number of the first series in the overlay chart and corresponds to the First Overlay Series box in the Overlay dialog box. If series_dist is 1 (automatic series distribution), this argument is ignored.

Up_down corresponds to the Up/Down Bars check box. Up_down is available only for line charts.

Series_line corresponds to the Series Lines check box. Series_line is available only for stacked bar and column charts.

Labels corresponds to the Radar Axis Labels check box. Labels is available only for radar charts.

Related Functions

DELETE.OVERLAY Deletes the overlay on a chart

FORMAT.CHART Formats a chart

FORMAT.SHAPE

Equivalent to clicking the reshape button on the Drawing toolbar and then inserting, moving, or deleting vertices of the selected polygon. A vertex is a point defined by a pair of coordinates in one row of the array that defines the polygon. The array is created by CREATE.OBJECT and EXTEND.POLYGON functions.

Syntax

FORMAT.SHAPE(**vertex_num, insert**, reference, x_offset, y_offset)

Vertex_num is a number corresponding to the vertex you want to insert, move, or delete.

Insert is a logical value specifying whether to insert a vertex, or move or delete a vertex.

- If insert is TRUE, Microsoft Excel inserts a vertex between the vertices vertex_num and vertex_num-1. The number of the new vertex then becomes vertex_num. The number of the vertex previously identified by vertex_num becomes vertex_num+1, and so on.
- If insert is FALSE, Microsoft Excel deletes the vertex (if the remaining arguments are omitted) or moves the vertex to the position specified by the remaining arguments.

Reference is the reference from which the vertex you are inserting or moving is measured; that is, the cell or range of cells to use as the basis for the x and y offsets.

- If reference is a range of cells, only the upper-left cell is used.
- If reference is omitted, the vertex is measured from the upper-left corner of the polygon's bounding rectangle.

X_offset is the horizontal distance from the upper-left corner of reference to the vertex. X_offset is measured in points. A point is 1/72nd of an inch. If reference is omitted, x_offset specifies the horizontal distance from the upper-left corner of the polygon bounding rectangle.

Y_offset is the vertical distance from the upper-left corner of reference to the vertex. Y_offset is measured in points. If reference is omitted, y_offset specifies the vertical distance from the upper-left corner of the polygon bounding rectangle.

Remarks

You cannot delete a vertex if only two vertices remain.

Examples

The following macro formula deletes the second vertex of the selected polygon:

```
FORMAT.SHAPE(2, FALSE)
```

The following macro formula moves the thirteenth vertex 6 points to the right and 4 points below the upper-left corner of cell B5 on the active worksheet:

```
FORMAT.SHAPE(13, FALSE, !$B$5, 6, 4)
```

The following macro formula inserts a new vertex between vertices 2 and 3. The new vertex is 60 points to the right and 75 points below the upper-left corner of the polygon's bounding rectangle:

```
FORMAT.SHAPE(3, TRUE, , 60, 75)
```

Related Functions

CREATE.OBJECT Creates an object

EXTEND.POLYGON Adds vertices to a polygon

FORMAT.SIZE

Equivalent to sizing an object with the mouse. Sizes the selected object and returns TRUE. If the selected chart object cannot be sized, FORMAT.SIZE returns FALSE. There are two syntax forms of this function. Use syntax 1 to size worksheet objects and chart items absolutely. Use syntax 2 relative to a cell or range of cells to size only worksheet objects. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax 1   Sizes worksheet objects and chart items

Syntax 2   Sizes worksheet objects relative to a cell or range

FORMAT.SIZE SYNTAX 1

Equivalent to sizing an object with the mouse. Sizes the selected object and returns TRUE. If the selected chart object cannot be sized, FORMAT.SIZE returns FALSE. There are two syntax forms of this function. Use syntax 1 to size worksheet objects and chart items absolutely. Use syntax 2 relative to a cell or range of cells to size only worksheet objects. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax

FORMAT.SIZE(width, height)

FORMAT.SIZE?(width, height)

Width   specifies the width of the selected object, measured in points. A point is 1/72nd of an inch.

Height   specifies the height of the selected object, measured in points.

You do not always have to use both arguments. For example, if you specify height and not width, the height changes but the width does not.

Remarks

- The base of a text label on a chart is the lower-left corner of the text rectangle.
- The base of an arrow is the end without the arrowhead.

Related Functions

FORMAT.MOVE   Moves the selected object

SIZE   Changes the size of a window

Syntax 2   Sizes worksheet objects relative to a cell or range

FORMAT.SIZE SYNTAX 2

Equivalent to sizing an object with the mouse. Sizes the selected worksheet object and returns TRUE. If the selected object cannot be sized, FORMAT.SIZE returns FALSE. There are two syntax forms of this function. Use syntax 2 to size worksheet objects relative to a cell or range of cells. Use syntax 1 to size worksheet objects and chart items. It is generally easier to use the macro recorder to enter this function on your macro sheet.

Syntax

FORMAT.SIZE(x_off, y_off, reference)

FORMAT.SIZE?(x_off, y_off, reference)

X_off specifies the width of the selected object and is measured in points from the lower-right corner of the object to the upper-left corner of reference. A point is 1/72nd of an inch. If omitted, x_off is assumed to be 0. If reference is omitted, x_off specifies the horizontal size.

Y_off specifies the height of the selected object and is measured in points from the lower-right corner of the object to the upper-left corner of reference. If omitted, y_off is assumed to be 0. If reference is omitted, y_off specifies the vertical size.

Reference specifies the cell or range of cells to use as the basis for the offset and for sizing. If reference is a range of cells, only the upper-left cell in the range is used.

Related Functions

FORMAT.MOVE Moves the selected object

SIZE Changes the size of a window

Syntax 1 Sizes worksheet objects and chart items

FORMAT.TEXT

Formats the selected worksheet text box or button or any text item on a chart.

Syntax

FORMAT.TEXT(x_align, y_align, orient_num, auto_text, auto_size, show_key, show_value, add_indent)

FORMAT.TEXT?(x_align, y_align, orient_num, auto_text, auto_size, show_key, show_value, add_indent)

Arguments correspond to check boxes or options in the various tabs on Format Object dialog box. Arguments that correspond to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box; if omitted, the current setting is used.

X_align is a number from 1 to 4 specifying the horizontal alignment of the text.

X_align	Horizontal alignment
1	Left
2	Center
3	Right

4	Justify

Y_align is a number from 1 to 4 specifying the vertical alignment of the text.

Y_align	**Vertical alignment**
1	Top
2	Center
3	Bottom
4	Justify

Orient_num is a number from 0 to 3 specifying the orientation of the text.

Orient_num	**Text orientation**
0	Horizontal
1	Vertical
2	Upward
3	Downward

Auto_text corresponds to the Automatic Text check box. If the selected text was created with the Data Labels command from the Insert menu and later edited, this option restores the original text. Auto_text is ignored for text boxes on worksheets and macro sheets.

Auto_size corresponds to the Automatic Size check box. If you have changed the size of the border around the selected text, this option restores the border to automatic size. Automatic size makes the border fit exactly around the text no matter how you change the text.

Show_key corresponds to the Show Legend Key Next to Label check box in the Data Labels dialog box. This argument applies only if the selected text is an attached data label on a chart.

Show_value corresponds to the Show Value option button in the Format Data Labels dialog box. This argument applies only if the selected text is an attached data label on a chart.

The following list summarizes which arguments apply to each type of text item.

Add_indent This argument is for only Far East versions of Microsoft Excel.

Text item	**Arguments that apply**
Worksheet text box or button	X_align, y_align, orient_num, auto_size
Attached data label	All arguments
Unattached text label	X_align, y_align, orient_num, auto_size
Tickmark label	Orient_num

Related Functions

CREATE.OBJECT Creates an object

FONT.PROPERTIES Applies a font to the selection

FORMULA Enters values into a cell or range or onto a chart

FORMULA

Enters a formula in the active cell or in a reference. There are two syntax forms of this function. Use syntax 1 to enter numbers, text, references, and formulas in a worksheet. Although syntax 1 can also be used to enter values on a macro sheet, you will not generally use FORMULA for this purpose. Use syntax 2 to enter a formula in a chart. For information about setting values on a macro sheet, see "Remarks" in the following topics.

Syntax 1 Enters numbers, text, references, and formulas in a worksheet

Syntax 2 Enters formulas in a chart

FORMULA SYNTAX 1

Enters a formula in the active cell or in a reference. If the active sheet is a worksheet, using FORMULA is equivalent to entering formula_text in the cell specified by reference. Formula_text is entered just as if you typed it in the formula bar.

There are two syntax forms of this function. Use syntax 1 to enter numbers, text, references, and formulas in a worksheet. Although syntax 1 can also be used to enter values on a macro sheet, you will not generally use FORMULA for this purpose. Use syntax 2 to enter a formula in a chart. For information about setting values on a macro sheet, see "Remarks" later in this topic.

Syntax

FORMULA(**formula_text**, reference)

Formula_text can be text, a number, a reference, or a formula in the form of text, or a reference to a cell containing any of the above.

- If formula_text contains references, they must be R1C1-style references, such as "=RC[1]*(1+R1C1)". If you are recording a macro when you enter a formula, Microsoft Excel converts A1-style references to R1C1-style references. For example, if you enter the formula =B2*(1+$A$1) in cell C2 while recording, Microsoft Excel records that action as =FORMULA("=RC[-1]*(1+R1C1)").

- If formula_text is a formula, the formula is entered. Text arguments must be surrounded by double sets of quotation marks. For example, to enter the formula =IF($A$1="Hello World", 1, 0) in the active cell with the FORMULA function, you would use the formula FORMULA("=IF(R1C1=""Hello World"", 1, 0)")

- If formula_text is a number, text, or logical value, the value is entered as a constant.

Reference specifies where formula_text is to be entered. It can be a reference to a cell in the active workbook or an external reference to a workbook. If reference is omitted, formula_text is entered in the active cell.

Remarks

Consider the following guidelines as you choose a function to set values on a worksheet or macro sheet:

- Use FORMULA to enter formulas and change values in a worksheet cell.

- SET.VALUE changes values on the macro sheet. Use SET.VALUE to assign initial values to a reference and to store values during the calculation of the macro.

- SET.NAME creates names on the macro sheet. Use SET.NAME to create a name and immediately assign a value to the name.

Examples

If the active sheet is a worksheet, the following macro formula enters the number constant 523 in the active cell:

```
FORMULA(523)
```

If the active sheet is a worksheet, the following macro formula enters the result of the INPUT function in cell A5:

```
FORMULA(INPUT("Enter a formula:", 0), !$A$5)
```

If you're using R1C1-style references and the active sheet is a worksheet, the following macro formula enters the formula =RC[-1]*(1+R1C1) in the active cell:

```
FORMULA("=RC[-1]*(1+R1C1)")
```

If the active sheet is a worksheet, the following macro formulas enter the number 1000 in the cell two rows down and three columns right from the active cell. The R1C1-style formula is shorter, but the OFFSET method may provide faster performance in larger macro sheets.

```
FORMULA(1000, OFFSET(ACTIVE.CELL(), 2, 3))

FORMULA(1000, "R[2]C[3]")
```

The following macro formula enters the phrase "Year to Date" in cell B4 on the sheet named SALES 1993:

```
FORMULA("Year to Date", 'SALES 1993'!B4)
```

Related Functions

FORMULA.ARRAY	Enters an array
FORMULA.FILL	Enters a formula in the specified range
SET.VALUE	Sets the value of a cell on a macro sheet
FORMULA Syntax 2	Enters formulas in a chart

FORMULA SYNTAX 2

Enters a text label or SERIES formula in a chart. To enter formulas on a worksheet or macro sheet, use syntax 1 of this function.

Syntax

FORMULA(**formula_text**)

Formula_text is the text label or SERIES formula you want to enter into the chart.

If	Then
Formula_text can be treated as a text label and the current selection is a text label	The selected text label is replaced with formula_text.
Formula_text can be treated as a text label and there is no current selection or the current selection is not a text label	Formula_text creates a new unattached text label.
Formula_text can be treated as a SERIES formula and the current selection is a SERIES formula	The selected SERIES formula is replaced with formula_text.
Formula_text can be treated as a SERIES formula and the current selection is not a SERIES formula	Formula_text creates a new SERIES formula.

Remarks

You would normally use the EDIT.SERIES function to create or edit a chart series. For more information, see EDIT.SERIES.

Example

The following macro formula enters a SERIES formula on the chart. If the current selection is a SERIES formula, it is replaced:

```
FORMULA("=SERIES(""Title"", , {1, 2, 3}, 1)")
```

Related Functions

EDIT.SERIES   Creates or changes a chart series

FORMULA, Syntax 1   Enters numbers, text, references, and formulas in a worksheet

FORMULA.ARRAY

Enters a formula as an array formula in the range specified or in the current selection. Equivalent to entering an array formula while pressing CTRL+SHIFT+ENTER in Microsoft Excel for Windows or COMMAND+ENTER in Microsoft Excel for the Macintosh.

Syntax

FORMULA.ARRAY(**formula_text**, reference)

Formula_text   is the text you want to enter in the array. For more information on formula_text, see the first form of FORMULA.

Reference   specifies where formula_text is entered. It can be a reference to a cell on the active worksheet or an external reference to a named workbook. Reference must be a R1C1-style reference in text form. If reference is omitted, formula_text is entered in the active cell.

Examples

If the selection is D25:E25, the following macro formula enters the array formula {=D22:E22+D23:E23} in the range D25:E25:

```
FORMULA.ARRAY("=R[-3]C:R[-3]C[1]+R[-2]C:R[-2]C[1]")
```

Regardless of the selection, the following macro formula enters the array formula {=D22:E22+D23:E23} in the range D25:E25:

```
FORMULA.ARRAY("=R[-3]C:R[-3]C[1]+R[-2]C:R[-2]C[1]", "R25C4:R25C5")
```

To use FORMULA.ARRAY to put an array in a specific workbook, specify the name of the workbook as an external reference in the reference argument. Using "[SALES.XLS]North!R25C3:R25C4" as the reference argument in the preceding example would enter the array in cells C25:D25 on the worksheet named North in the workbook SALES.XLS. Using "SALES!R25C3:R25C4" as the reference argument would enter the array in the same cells in the worksheet named SALES.

Related Functions

FORMULA   Enters values into a cell or range or onto a chart

FORMULA.FILL   Enters a formula in the specified range

FORMULA.CONVERT

Changes the style and type of references in a formula between A1 and R1C1 and between relative and absolute. Use FORMULA.CONVERT to convert references of one style or type to another style or type.

Syntax

FORMULA.CONVERT(**formula_text, from_a1**, to_a1, to_ref_type, rel_to_ref)

Formula_text is the formula, given as text, containing the references you want to change. Formula_text must be a valid formula, and an equal sign must be included.

From_a1 is a logical value specifying whether the references in formula_text are in A1 or R1C1 style. If from_a1 is TRUE, references are in A1 style; if FALSE, references are in R1C1 style.

To_a1 is a logical value specifying the form for the references FORMULA.CONVERT returns. If to_a1 is TRUE, references are returned in A1 style; if FALSE, references are returned in R1C1 style. If to_a1 is omitted, the reference style is not changed.

To_ref_type is a number from 1 to 4 specifying the reference type of the returned formula. If to_ref_type is omitted, the reference type is not changed.

To_ref_type	**Reference type returned**
1	Absolute
2	Absolute row, relative column
3	Relative row, absolute column
4	Relative

Rel_to_ref is an absolute reference that specifies what cell the relative references are or should be relative to.

Examples

Use FORMULA.CONVERT to convert relative references entered by the user in an INPUT function or custom dialog box into absolute references. The following macro formula converts the given formula to an absolute, R1C1-style reference:

`FORMULA.CONVERT("=A1:A10", TRUE, FALSE, 1)` equals "=R1C1:R10C1"

The following macro formula converts the references in the given formula to relative, A1-style references:

`FORMULA.CONVERT("=SUM(R10C2:R15C2)", FALSE, TRUE, 4)` equals "=SUM(B10:B15)"

Tip To put the converted formula into a cell or range of cells, use the FORMULA.CONVERT function as the formula_text argument to the FORMULA function.

Related Functions

ABSREF Returns the absolute reference of a range of cells to another range

FORMULA Enters values into a cell or range or onto a chart

RELREF Returns a relative reference

FORMULA.FILL

Enters a formula in the range specified or in the current selection. Equivalent to entering a formula in a range of cells while pressing CTRL+ENTER in Microsoft Excel for Windows or OPTION+ENTER in Microsoft Excel for the Macintosh.

Syntax

FORMULA.FILL(**formula_text**, reference)

Formula_text is the text with which you want to fill the range. For more information on formula_text, see FORMULA.

Reference specifies where formula_text is entered. It can be a reference to a range in the active worksheet or an external reference to a named workbook. If omitted, formula_text is entered in the current selection.

Related Functions

DATA.SERIES Fills a range of cells with a series of numbers or dates

FORMULA Enters values into a cell or range or onto a chart

FORMULA.ARRAY Enters an array

FORMULA.FIND

Equivalent to clicking the Find command on the Edit menu. Selects the next or previous cell containing the specified text and returns TRUE. If a matching cell is not found, FORMULA.FIND returns FALSE and displays a message.

Syntax

FORMULA.FIND(**text, in_num, at_num, by_num**, dir_num, match_case)

FORMULA.FIND?(text, in_num, at_num, by_num, dir_num, match_case)

Text is the text you want to find. Text corresponds to the Find What box in the Find dialog box.

In_num is a number from 1 to 3 specifying where to search.

In_num	Searches
1	Formulas

2	Values
3	Notes

At_num is the number 1 or 2 and specifies whether to find cells containing only text or also cells containing text within a longer string of characters.

At_num	Searches for text as
1	A whole string (the only value in the cell)
2	Either a whole string or part of a longer string

By_num is the number 1 or 2 and specifies whether to search by rows or by columns.

By_num	Searches by
1	Rows
2	Columns

Dir_num is the number 1 or 2 and specifies whether to search for the next or previous occurrence of text.

Dir_num	Searches for
1 or omitted	The next occurrence of text
2	The previous occurrence of text

Match_case is a logical value corresponding to the Match Case check box in the Find dialog box. If match_case is TRUE, Microsoft Excel matches characters exactly, including uppercase and lowercase; if FALSE or omitted, matching is not case-sensitive.

Remarks

- In Microsoft Excel for Windows, the dialog-box form of FORMULA.FIND is equivalent to pressing SHIFT+F5.

- If more than one cell is selected when you use FORMULA.FIND, Microsoft Excel searches only that selection.

FORMULA.FIND.NEXT, FORMULA.FIND.PREV

Finds the next and previous cells on the worksheet, as specified in the Find dialog box, and returns TRUE. (To see the Find dialog box, click Find on the Edit menu.) If a matching cell is not found, the functions return FALSE. For more information see FORMULA.FIND.

Syntax

FORMULA.FIND.NEXT( )

FORMULA.FIND.PREV( )

Related Functions

DATA.FIND	Selects records in a database that match the specified criteria
FORMULA.FIND	Finds text in a workbook

FORMULA.GOTO

Equivalent to clicking the Go To command on the Edit menu or to pressing F5. Scrolls through the worksheet and selects a named area or reference. Use FORMULA.GOTO to select a range on any open workbook; use SELECT to select a range on the active workbook.

Syntax

FORMULA.GOTO(reference, corner)

FORMULA.GOTO?(reference, corner)

Reference specifies where to scroll and what to select.

- Reference should be either an external reference to a workbook, an R1C1-style reference in the form of text (see the second example following), or a name.
- If the Go To command has already been carried out, reference is optional. If reference is omitted, it is assumed to be the reference of the cells you selected before the previous Go To command or FORMULA.GOTO macro function was carried out. This feature distinguishes FORMULA.GOTO from SELECT.

Corner is a logical value that specifies whether to scroll through the window so that the upper-left cell in reference is in the upper-left corner of the active window. If corner is TRUE, Microsoft Excel places reference in the upper-left corner of the window; if FALSE or omitted, Microsoft Excel scrolls through normally.

Tip Microsoft Excel keeps a list of the cells you've selected with previous FORMULA.GOTO functions or Go To commands. When you use FORMULA.GOTO with GET.WORKSPACE(41), which returns a horizontal array of previous Go To selections, you can backtrack through multiple previous selections. See the last example below.

Remarks

- If you are recording a macro when you click the Go To command, the reference you enter in the Reference box of the Go To dialog box is recorded as text in the R1C1 reference style.

- If you are recording a macro when you double-click a cell that has precedents on another worksheet, Microsoft Excel records a FORMULA.GOTO function.

Examples

Each of the following macro formulas goes to cell A1 on the active worksheet:

```
FORMULA.GOTO(!$A$1)
```

```
FORMULA.GOTO("R1C1")
```

Each of the following macro formulas goes to the cells named Sales on the active worksheet and scrolls through the worksheet so that the upper-left corner of Sales is in the upper-left corner of the window:

```
FORMULA.GOTO(!Sales, TRUE)
```

```
FORMULA.GOTO("Sales", TRUE)
```

The following macro formula goes to the cells that were selected by the third most recent FORMULA.GOTO function or Go To command:

```
FORMULA.GOTO(INDEX(GET.WORKSPACE(41), 1, 3))
```

Related Functions

GOTO   Directs macro execution to another cell

HSCROLL   Horizontally scrolls through a sheet by percentage or by column or row number

SELECT   Selects a cell, worksheet object, or chart item

VSCROLL   Vertically scrolls through a sheet by percentage or by column or row number

FORMULA.REPLACE

Equivalent to clicking the Replace command on the Edit menu. Finds and replaces characters in cells on your worksheet.

Syntax

FORMULA.REPLACE(**find_text, replace_text**, look_at, look_by, active_cell, match_case)

FORMULA.REPLACE?(find_text, replace_text, look_at, look_by, active_cell, match_case)

Find_text   is the text you want to find. You can use the wildcard characters, question mark (?) and asterisk (*), in find_text. A question mark matches any single character; an asterisk matches any sequence of characters. If you want to find an actual question mark or asterisk, type a tilde (~) before the character.

Replace_text   is the text you want to replace find_text with.

Look_at is a number specifying whether you want find_text to match the entire contents of a cell or any string of matching characters.

Look_at	Looks for find_text
1 or omitted	As the entire contents of a cell
2	As part of the contents of a cell

Look_by is a number specifying whether to search horizontally (through rows) or vertically (through columns).

Look_by	Looks for find_text
1 or omitted	By rows
2	By columns

Active_cell is a logical value specifying the cells in which find_text is to be replaced.

- If active_cell is TRUE, find_text is replaced in the active cell only.
- If active_cell is FALSE, find_text is replaced in the entire selection, or, if the selection is a single cell, in the entire sheet.

Match_case is a logical value corresponding to the Match Case check box in the Replace dialog box. If match_case is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box. If match_case is omitted, the status of the check box is unchanged.

Remarks

- In FORMULA.REPLACE?, the dialog-box form of the function, omitted arguments are assumed to be the same arguments used in the previous replace operation. If there was no previous replace operation, omitted text arguments are assumed to be "" (empty text).
- The result of FORMULA.REPLACE must be a valid cell entry. For example, you cannot replace "=" with "= =" at the beginning of a formula.
- If more than a single cell is selected before you use FORMULA.REPLACE, only the selected cells are searched.

Related Function

FORMULA.FIND Finds text in a workbook

FOURIER

Performs a Fourier transform.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

FOURIER(**inprng**, outrng, inverse, labels)

FOURIER?(inprng, outrng, inverse, labels)

Inprng is the input range. The number of cells in the input range must be equal to a power of two (2, 4, 8, 16, ...).

Outrng is the first cell in the output range or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Inverse is a logical value. If TRUE, an inverse Fourier transform is performed. If FALSE or omitted, a forward Fourier transform is performed.

Labels is a logical value.

- If labels is TRUE, then the first row or column of inprng contains labels.
- If labels is FALSE or omitted, all cells in inprng are considered data. Microsoft Excel generates appropriate data labels for the output table.

Related Function

SAMPLE Samples data

FPOS

Sets the position of a file. The position of a file is where a character is read from or written to by an FREAD, FREADLN, FWRITE, or FWRITELN function. Use FPOS when you want to write characters to or read characters from specific locations. For example, to append text to the end of a file, you must set the position to the end of the file; otherwise, you might accidentally overwrite existing characters in the file.

Syntax

FPOS(**file_num**, position_num)

File_num is the unique ID number of the file for which you want to set the position. File_num is returned by a previously executed FOPEN function. If file_num is not valid, FPOS returns the #VALUE! error value.

Position_num is the location in the file that a character will be read from or written to.

- The first position in a file is 1, the location of the first byte.
- The last position in the file is the same as the value returned by FSIZE. For example, the last position in a file with 280 bytes is 280.

- If position_num is omitted, FPOS returns the current position of the file—that is, the number corresponding to where the next character will be read from or written to.

Whenever you read a character from or write a character to a file, the file's position is automatically incremented.

Examples

The following statement starts a loop that executes until the position in the open file identified as FileNumber reaches the end of the file:

```
=WHILE(FPOS(FileNumber)<=FSIZE(FileNumber))
```

Related Functions

FCLOSE	Closes a text file
FOPEN	Opens a file with the type of permission specified
FREAD	Reads characters from a text file
FREADLN	Reads a line from a text file
FWRITE	Writes characters to a text file
FWRITELN	Writes a line to a text file

FREAD

Reads characters from a file, starting at the current position in the file. (For more information about a file's position, see FPOS.) If FREAD is successful, it returns the text to the cell containing FREAD and set's the file's position to the start of the following line. If the end of the file is reached or if FREAD can't read the file, it returns the #N/A error value. Use FREAD instead of FREADLN when you need to read a specific number of characters from a text file.

Syntax

FREAD(**file_num, num_chars**)

File_num is the unique ID number of the file you want to read data from. File_num is returned by a previously executed FOPEN function. If file_num is not valid, FREAD returns the #VALUE! error value.

Num_chars specifies how many bytes to read from the file. FREAD can read up to 255 bytes at a time.

Example

The following function reads the next 200 bytes from the open file identified as FileNumber:

```
FREAD(FileNumber, 200)
```

Related Functions

FOPEN	Opens a file with the type of permission specified
FPOS	Sets the position in a text file

FREADLN   Reads a line from a text file

FWRITE   Writes characters to a text file

FREADLN

Reads characters from a file, starting at the current position in the file and continuing to the end of the line, placing the characters in the cell containing FREADLN. (For more information about a file's position, see FPOS.) If FREADLN is successful, it returns the text it read, up to but not including the carriage-return and linefeed characters at the end of the line (in Microsoft Excel for Windows) or the carriage-return character at the end of the line (in Microsoft Excel for the Macintosh). If the current file position is the end of the file or if FREADLN can't read the file, it returns the #N/A error value.

Syntax

FREADLN(**file_num**)

File_num   is the unique ID number of the file you want to read data from. File_num is returned by a previously executed FOPEN function. If file_num is not valid, FREADLN returns the #VALUE! error value.

Example

The following function reads the next line from the open file identified as FileNumber:

```
FREADLN(FileNumber)
```

Related Functions

FOPEN   Opens a file with the type of permission specified

FPOS   Sets the position in a text file

FREAD   Reads characters from a text file

FWRITE   Writes characters to a text file

FWRITELN   Writes a line to a text file

FREEZE.PANES

Equivalent to clicking the Freeze Panes or Unfreeze Panes command on the Window menu. Splits the active window into panes, creates frozen panes, or freezes or unfreezes existing panes. Use FREEZE.PANES to keep row or column titles on the screen while scrolling to other parts of the sheet.

Syntax

FREEZE.PANES(logical, col_split, row_split)

Logical   is a logical value specifying which command FREEZE.PANES is equivalent to.

- If logical is TRUE, the function is equivalent to the Freeze Panes command. It freezes panes if they exist, or creates them, splits them at the specified position, and freezes them if they do not exist. If the panes are already frozen, FREEZE.PANES takes no action.

- If logical is FALSE, the function is equivalent to the Unfreeze Panes command. If no panes exist, FREEZE.PANES takes no action.
- If logical is omitted, FREEZE.PANES creates and then freezes panes if no panes exist, freezes existing panes if they're not currently frozen, or unfreezes existing panes if they're currently frozen.

Col_split specifies where to split the window vertically and is measured in columns from the left of the window.

Row_split specifies where to split the window horizontally and is measured in rows from the top of the window.

Col_split and row_split are ignored unless logical is TRUE and split panes do not exist.

Remarks

To create panes without freezing or unfreezing them, use the SPLIT function. You can freeze the panes later using the FREEZE.PANES function.

Related Functions

ACTIVATE	Switches to a window
SPLIT	Splits a window

FSIZE

Returns the number of bytes in a file. Use FSIZE to determine the size of the file, which is the same as the position of the last byte in the file.

Syntax

FSIZE(**file_num**)

File_num is the unique ID number of the file whose size you want to know. File_num is returned by a previously executed FOPEN function. If file_num is not valid, FSIZE returns the #VALUE! error value.

Example

The following function returns the size in bytes of the open file identified as FileNumber:

```
FSIZE(FileNumber)
```

Related Functions

FOPEN	Opens a file with the type of permission specified
FPOS	Sets the position in a text file

FTESTV

Performs a two-sample F-test.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

FTESTV(**inprng1, inprng2**, outrng, labels)

FTESTV?(inprng1, inprng2. outrng, labels)

Inprng1 is the input range for the first data set.

Inprng2 is the input range for the second data set.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Labels is a logical value.

- If labels is TRUE, then the first row or column of inprng1 and inprng2 contain labels.
- If labels is FALSE or omitted, all cells in inprng1 and inprng2 are considered data. Microsoft Excel generates appropriate data labels for the output table.

FULL

Equivalent to pressing CTRL+F10 (full size) and CTRL+F5 (previous size) or double-clicking the title bar in Microsoft Excel for Windows version 3.0 or earlier. Equivalent to double-clicking the title bar or clicking the zoom box in Microsoft Excel for the Macintosh version 3.0 or earlier. This function is included only for macro compatibility. To perform the equivalent of a FULL(TRUE) function in Microsoft Excel version 4.0 or later, use the WINDOW.MAXIMIZE function. To perform the equivalent of a FULL(FALSE) function in Microsoft Excel version 4.0 or later, use the WINDOW.RESTORE function.

Syntax

FULL(logical)

FULL.SCREEN

Equivalent to clicking the Full Screen command on the View menu.

Syntax

FULL.SCREEN(logical)

Logical switches to full screen if TRUE or omitted; exits full screen mode if FALSE.

FUNCTION.WIZARD

Displays the Paste Function dialog box, which you can use to enter functions into cells.

Syntax

FUNCTION.WIZARD?( )

Remarks

If you know the function or formula that you want to insert into a cell, use the FORMULA function.

Related Function

FORMULA Enters values into a cell or range or onto a chart

FWRITE

Writes text to a file, starting at the current position in that file. (For more information about a file's position, see FPOS.) If FWRITE can't write to the file, it returns the #N/A error value.

Syntax

FWRITE(**file_num, text**)

File_num is the unique ID number of the file you want to write data to. File_num is returned by a previously executed FOPEN function. If file_num is not valid, FWRITE returns the #VALUE! error value.

Text is the text you want to write to the file.

Example

The following function writes the current month to the open file identified as FileNumber:

```
FWRITE(FileNumber, TEXT(MONTH(NOW()),"mmmm"))
```

Related Functions

FOPEN Opens a file with the type of permission specified

FPOS Sets the position in a text file

FREAD Reads characters from a text file

FWRITELN Writes a line to a text file

FWRITELN

Writes text, followed by a carriage return and linefeed, to a file, starting at the current position in that file. (For more information about a file's position, see FPOS.) If FWRITELN can't write to the file, it returns the #N/A error value. Use FWRITELN instead of FWRITE when you want to append a carriage return and linefeed to each group of characters that you write to a text file.

Syntax

FWRITELN(**file_num, text**)

File_num is the unique ID number of the file you want to write data to. File_num is returned by a previously executed FOPEN function. If file_num is not valid, FWRITELN returns the #VALUE! error value.

Text is the text you want to write to the file.

Remarks

In Microsoft Excel for Windows, FWRITELN writes text followed by a carriage return and a line feed. In Microsoft Excel for the Macintosh, FWRITELN writes text followed by a carriage return only.

Example

The following function writes the current month to the open file identified as FileNumber and starts a new line in the file:

```
FWRITELN(FileNumber, TEXT(MONTH(NOW()),"mmmm"))
```

Related Functions

FOPEN	Opens a file with the type of permission specified
FPOS	Sets the position in a text file
FREAD	Reads characters from a text file
FWRITE	Writes characters to a text file

GALLERY.3D.AREA

Changes the format of the active chart to a 3-D area chart.

Syntax

GALLERY.3D.AREA(**type_num**)

GALLERY.3D.AREA?(type_num)

Type_num is the number of the 3-D Area format that you want to apply to the chart.

GALLERY.3D.BAR

Changes the active chart to a 3-D bar chart.

Syntax

GALLERY.3D.BAR(**type_num**)

GALLERY.3D.BAR?(type_num)

Type_num is the number of the 3-D Bar format that you want to apply to the chart.

GALLERY.3D.COLUMN

Changes the format of the active chart to a 3-D column chart.

Syntax

GALLERY.3D.COLUMN(**type_num**)

GALLERY.3D.COLUMN?(type_num)

Type_num is the number of the 3-D Column format that you want to apply to the chart.

GALLERY.3D.LINE

Changes the format of the active chart to a 3-D line chart.

Syntax

GALLERY.3D.LINE(**type_num**)

GALLERY.3D.LINE?(type_num)

Type_num is the number of the 3-D Line format that you want to apply to the chart.

GALLERY.3D.PIE

Changes the format of the active chart to a 3-D pie chart.

Syntax

GALLERY.3D.PIE(**type_num**)

GALLERY.3D.PIE?(type_num)

Type_num is the number of the 3-D Pie format that you want to apply to the chart.

GALLERY.3D.SURFACE

Changes the active chart to a 3-D surface chart.

Syntax

GALLERY.3D.SURFACE(**type_num**)

GALLERY.3D.SURFACE?(type_num)

Type_num is the number of the 3-D Surface format that you want to apply to the chart.

GALLERY.AREA

Changes the format of the active chart to an area chart.

Syntax

GALLERY.AREA(**type_num**, delete_overlay)

GALLERY.AREA?(type_num, delete_overlay)

Type_num is the number of a format in the AutoFormat dialog box when a chart is active dialog box that you want to apply to the area chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.BAR

Changes the format of the active chart to a bar chart.

Syntax

GALLERY.BAR(**type_num**, delete_overlay)

GALLERY.BAR?(type_num, delete_overlay)

Type_num is the number of the format that you want to apply to the bar chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.COLUMN

Changes the format of the active chart to a column chart.

Syntax

GALLERY.COLUMN(**type_num**, delete_overlay)

GALLERY.COLUMN?(type_num, delete_overlay)

Type_num is the number of the format you want to apply to the column chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.CUSTOM

Changes the format of the active chart to the custom format.

Syntax

GALLERY.CUSTOM(**name_text**)

Name_text is the name of the custom template you want to apply.

Related Functions

ADD.CHART.AUTOFORMAT Formats a chart using a custom gallery

DELETE.CHART.AUTOFORMAT Deletes a custom gallery

GALLERY.DOUGHNUT

Changes the format of the active chart to a doughnut chart.

GALLERY.DOUGHNUT(**type_num**, delete_overlay)

GALLERY.DOUGHNUT?(type_num, delete_overlay)

Type_num is the number of the format you want to apply to the doughnut chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.LINE

Changes the format of the active chart to a line chart.

Syntax

GALLERY.LINE(**type_num**, delete_overlay)

GALLERY.LINE?(type_num, delete_overlay)

Type_num is the number of the format you want to apply to the line chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.PIE

Changes the format of the active chart to a pie chart.

Syntax

GALLERY.PIE(**type_num**, delete_overlay)

GALLERY.PIE?(type_num, delete_overlay)

Type_num is the number of the format you want to apply to the pie chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.RADAR

Changes the format of the active chart to a radar chart.

Syntax

GALLERY.RADAR(**type_num**, delete_overlay)

GALLERY.RADAR?(type_num, delete_overlay)

Type_num is the number of the format you want to apply to the radar chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GALLERY.SCATTER

Changes the format of the active chart to an xy (scatter) chart.

Syntax

GALLERY.SCATTER(**type_num**, delete_overlay)

GALLERY.SCATTER?(type_num, delete_overlay)

Type_num is the number of the format you want to apply to the xy (scatter) chart.

Delete_overlay is a logical value specifying whether to delete an overlay chart.

- If delete_overlay is TRUE, Microsoft Excel deletes all overlays, if present, and applies the new format to the main chart.
- If delete_overlay is FALSE or omitted, Microsoft Excel applies the new format to either the main chart or the overlay, depending on the location of the selected series.

GET.BAR

Returns the number of the active menu bar. There are two syntax forms of GET.BAR. Use syntax 1 to return information that you can use with other functions that manipulate menu bars. Use syntax 2 to return information that you can use with functions that add, delete, or alter menu commands.

Syntax 1   Returns the number of the active menu bar

Syntax 2   Returns the name or position number of a specified command on a menu or of a specified menu on a menu bar

GET.BAR SYNTAX 1

Returns the number of the active menu bar. There are two syntax forms of GET.BAR. Use syntax 1 to return information that you can use with other functions that manipulate menu bars. For a list of the ID numbers for Microsoft Excel's built-in menu bars, see ADD.COMMAND.

Syntax

GET.BAR( )

Example

The following macro formula assigns the name OldBar to the number of the active menu bar. This is useful if you will need to restore the current menu bar after displaying another custom menu bar.

```
SET.NAME("OldBar", GET.BAR())
```

Related Functions

ADD.BAR   Adds a menu bar

SHOW.BAR   Displays a menu bar

GET.BAR Syntax 2   Returns the name or position number of a specified command on a menu or of a specified menu on a menu bar

GET.BAR SYNTAX 2

Returns the name or position number of a specified command on a menu or of a specified menu on a menu bar. There are two syntax forms of GET.BAR. Use syntax 2 to return information that you can use with functions that add, delete, or alter menu commands.

Syntax

GET.BAR(**bar_num, menu, command**, subcommand)

Bar_num is the number of a menu bar containing the menu or command about which you want information. Bar_num can be the number of a built-in menu bar or the number returned by a previously run ADD.BAR function. For a list of the ID numbers for Microsoft Excel's built-in menu bars, see ADD.COMMAND.

Menu is the menu on which the command resides or the menu whose name or position you want. Menu can be the name of the menu as text or the number of the menu. Menus are numbered starting with 1 from the left of the menu bar.

Command is the command or submenu whose name or number you want returned. Command can be the name of the command from the menu as text, in which case the number is returned, or the number of the command from the menu, in which case the name is returned. Commands are numbered starting with 1 from the top of the menu. If command is 0, the name or position number of the menu is returned. If an ellipsis (...) follows a command name, such as the Open... command on the File menu, then you must include the ellipsis when referring to that command. See the following examples.

Subcommand returns the name (if number is used for subcommand) or position (if name is used for subcommand) of a command on a submenu. If the command argument refers to an empty submenu, or is a command instead of a submenu, then using subcommand returns #N/A.

Remarks

- If an ampersand is used to indicate the access key in the name of a custom command, the ampersand is included in the name returned by GET.BAR. All built-in commands have an ampersand before the letter used as the access key.

- If the command name or position specified does not exist, GET.BAR returns the #N/A error value.

Examples

In the default worksheet and macro sheet menu bar:

`GET.BAR(10, "File", "Print...")` equals 14

`GET.BAR(10, "File", 14)` equals "&Print...^tCTRL+P" (where ^t is a tab character)

`GET.BAR(10, 1, "Open")` equals #N/A

`GET.BAR(10, 1, "Open...")` equals 2

Related Functions

ADD.COMMAND	Adds a command to a menu
DELETE.COMMAND	Deletes a command from a menu
GET.TOOLBAR	Retrieves information about a toolbar
RENAME.COMMAND	Changes the name of a command or menu
GETBAR Syntax 1	Returns the number of the active menu bar

GET.CELL

Returns information about the formatting, location, or contents of a cell. Use GET.CELL in a macro whose behavior is determined by the status of a particular cell.

Syntax

GET.CELL(**type_num**, reference)

Type_num is a number that specifies what type of cell information you want. The following list shows the possible values of type_num and the corresponding results.

Type_num	**Returns**
1	Absolute reference of the upper-left cell in reference, as text in the current workspace reference style.
2	Row number of the top cell in reference.
3	Column number of the leftmost cell in reference.
4	Same as TYPE(reference).
5	Contents of reference.
6	Formula in reference, as text, in either A1 or R1C1 style depending on the workspace setting.
7	Number format of the cell, as text (for example, "m/d/yy" or "General").
8	Number indicating the cell's horizontal alignment: 1 = General 2 = Left 3 = Center 4 = Right 5 = Fill 6 = Justify 7 = Center across cells
9	Number indicating the left-border style assigned to the cell:

	0 = No border 1 = Thin line 2 = Medium line 3 = Dashed line 4 = Dotted line 5 = Thick line 6 = Double line 7 = Hairline
10	Number indicating the right-border style assigned to the cell. See type_num 9 for descriptions of the numbers returned.
11	Number indicating the top-border style assigned to the cell. See type_num 9 for descriptions of the numbers returned.
12	Number indicating the bottom-border style assigned to the cell. See type_num 9 for descriptions of the numbers returned.
13	Number from 0 to 18, indicating the pattern of the selected cell as displayed in the Patterns tab of the Format Cells dialog box, which appears when you click the Cells command on the Format menu. If no pattern is selected, returns 0.
14	If the cell is locked, returns TRUE; otherwise, returns FALSE.
15	If the cell's formula is hidden, returns TRUE; otherwise, returns FALSE.
16	A two-item horizontal array containing the width of the active cell and a logical value indicating whether the cell's width is set to change as the standard width changes (TRUE) or is a custom width (FALSE).
17	Row height of cell, in points.
18	Name of font, as text.
19	Size of font, in points.
20	If all the characters in the cell, or only the first character, are bold, returns TRUE; otherwise, returns FALSE.
21	If all the characters in the cell, or only the first character, are italic, returns TRUE; otherwise, returns FALSE.

22	If all the characters in the cell, or only the first character, are underlined, returns TRUE; otherwise, returns FALSE.
23	If all the characters in the cell, or only the first character, are struck through, returns TRUE; otherwise, returns FALSE.
24	Font color of the first character in the cell, as a number in the range 1 to 56. If font color is automatic, returns 0.
25	If all the characters in the cell, or only the first character, are outlined, returns TRUE; otherwise, returns FALSE. Outline font format is not supported by Microsoft Excel for Windows.
26	If all the characters in the cell, or only the first character, are shadowed, returns TRUE; otherwise, returns FALSE. Shadow font format is not supported by Microsoft Excel for Windows.
27	Number indicating whether a manual page break occurs at the cell: 0 = No break 1 = Row 2 = Column 3 = Both row and column
28	Row level (outline)
29	Column level (outline).
30	If the row containing the active cell is a summary row, returns TRUE; otherwise, returns FALSE.
31	If the column containing the active cell is a summary column, returns TRUE; otherwise, returns FALSE.
32	Name of the workbook and sheet containing the cell If the window contains only a single sheet that has the same name as the workbook without its extension, returns only the name of the book, in the form BOOK1.XLS. Otherwise, returns the name of the sheet in the form "[Book1]Sheet1".
33	If the cell is formatted to wrap, returns TRUE; otherwise, returns FALSE.
34	Left-border color as a number in the range 1 to 56. If color is automatic, returns 0.
35	Right-border color as a number in the range 1 to 56. If color is automatic, returns 0.

36	Top-border color as a number in the range 1 to 56. If color is automatic, returns 0.
37	Bottom-border color as a number in the range 1 to 56. If color is automatic, returns 0.
38	Shade foreground color as a number in the range 1 to 56. If color is automatic, returns 0.
39	Shade background color as a number in the range 1 to 56. If color is automatic, returns 0.
40	Style of the cell, as text.
41	Returns the formula in the active cell without translating it (useful for international macro sheets).
42	The horizontal distance, measured in points, from the left edge of the active window to the left edge of the cell. May be a negative number if the window is scrolled beyond the cell.
43	The vertical distance, measured in points, from the top edge of the active window to the top edge of the cell. May be a negative number if the window is scrolled beyond the cell.
44	The horizontal distance, measured in points, from the left edge of the active window to the right edge of the cell. May be a negative number if the window is scrolled beyond the cell.
45	The vertical distance, measured in points, from the top edge of the active window to the bottom edge of the cell. May be a negative number if the window is scrolled beyond the cell.
46	If the cell contains a text note, returns TRUE; otherwise, returns FALSE.
47	If the cell contains a sound note, returns TRUE; otherwise, returns FALSE.
48	If the cells contains a formula, returns TRUE; if a constant, returns FALSE.
49	If the cell is part of an array, returns TRUE; otherwise, returns FALSE.
50	Number indicating the cell's vertical alignment: 1 = Top 2 = Center

	3 = Bottom 4 = Justified
51	Number indicating the cell's vertical orientation: 0 = Horizontal 1 = Vertical 2 = Upward 3 = Downward
52	The cell prefix (or text alignment) character, or empty text ("") if the cell does not contain one.
53	Contents of the cell as it is currently displayed, as text, including any additional numbers or symbols resulting from the cell's formatting.
54	Returns the name of the PivotTable report containing the active cell.
55	Returns the position of a cell within the PivotTable report. 0 = Row header 1 = Column header 2 = Page header 3 = Data header 4 = Row item 5 = Column item 6 = Page item 7 = Data item 8 = Table body
56	Returns the name of the field containing the active cell reference if inside a PivotTable report.
57	Returns TRUE if all the characters in the cell, or only the first character, are formatted with a superscript font; otherwise, returns FALSE.
58	Returns the font style as text of all the characters in the cell, or only the first character as displayed in the Font tab of the Format Cells dialog box: for example, "Bold Italic".
7	Returns the number for the underline style:

	1 = None 2 = Single 3 = Double 4 = Single accounting 5 = Double accounting
60	Returns TRUE if all the characters in the cell, or only the first character, are formatted with a subscript font; otherwise, it returns FALSE.
61	Returns the name of the PivotTable item for the active cell, as text.
62	Returns the name of the workbook and the current sheet in the form "[Book1]Sheet1".
63	Returns the fill (background) color of the cell.
64	Returns the pattern (foreground) color of the cell.
65	Returns TRUE if the Add Indent alignment option is on (Far East versions of Microsoft Excel only); otherwise, it returns FALSE.
66	Returns the book name of the workbook containing the cell in the form BOOK1.XLS.

Reference is a cell or a range of cells from which you want information.

- If reference is a range of cells, the cell in the upper-left corner of the first range in reference is used.
- If reference is omitted, the active cell is assumed.

Tip Use GET.CELL(17) to determine the height of a cell and GET.CELL(44) - GET.CELL(42) to determine the width.

Examples

The following macro formula returns TRUE if cell B4 on sheet Sheet1 is bold:

```
GET.CELL(20, Sheet1!$B$4)
```

You can use the information returned by GET.CELL to initiate an action. The following macro formula runs a custom function named BoldCell if the GET.CELL formula returns FALSE:

```
IF(GET.CELL(20, Sheet1!$B$4), , BoldCell())
```

Related Functions

ABSREF   Returns the absolute reference of a range of cells to another range

ACTIVE.CELL   Returns the reference of the active cell

GET.FORMULA   Returns the contents of a cell

GET.NAME   Returns the definition of a name

GET.NOTE   Returns characters from a note

RELREF   Returns a relative reference

GET.CHART.ITEM

Returns the vertical or horizontal position of a point on a chart item. Use these position numbers with FORMAT.MOVE and FORMAT.SIZE to change the position and size of chart items. Position is measured in points; a point is 1/72nd of an inch.

Syntax

GET.CHART.ITEM(**x_y_index**, point_index, item_text)

X_y_index   is a number specifying which of the coordinates you want returned.

X_y_index	Coordinate returned
1	Horizontal coordinate
2	Vertical coordinate

Point_index   is a number specifying the point on the chart item. These indexes are described below. If point_index is omitted, it is assumed to be 1.

- If the specified item is a point, point_index must be 1.
- If the specified item is any line other than a data line, use the following values for point_index.

Point_index	Chart item position
1	Lower or left
2	Upper or right

- If the selected item is a legend, plot area, chart area, or an area in an area chart, use the following values for point_index.

Point_index	**Chart item position**
1	Upper left
2	Upper middle
3	Upper right
4	Right middle
5	Lower right
6	Lower middle
7	Lower left
8	Left middle

- If the selected item is an arrow in Microsoft Excel 4.0, use the following values for point_index. In Microsoft Excel version 5.0 or later, arrows are named lines, and the arrowhead position returned is equivalent to the end of a line where the arrowhead begins.

Point_index	**Chart item position**
1	Arrow shaft
2	Arrowhead

- If the selected item is a pie slice, use the following values for point_index.

Point_index	Chart item position
1	Outermost counterclockwise point
2	Outer center point
3	Outermost clockwise point
4	Midpoint of the most clockwise radius
5	Center point
6	Midpoint of the most counterclockwise radius

Item_text is a selection code that specifies which item of a chart to select. See the chart form of SELECT for the item_text codes to use for each item of a chart.

- If item_text is omitted, it is assumed to be the currently selected item.
- If item_text is omitted and no item is selected, GET.CHART.ITEM returns the #VALUE! error value.

Remarks

If the specified item does not exist, or if a chart is not active when the function is carried out, the #VALUE! error value is returned.

Examples

The following macro formulas return the horizontal and vertical locations, respectively, of the top of the main-chart value axis:

```
GET.CHART.ITEM(1, 2, "Axis 1")
```

```
GET.CHART.ITEM(2, 2, "Axis 1")
```

You could then use FORMAT.MOVE to move a floating text item to the position returned by these two formulas.

Related Functions

GET.DOCUMENT Returns information about a workbook

GET.FORMULA Returns the contents of a cell

GET.DEF

Returns the name, as text, that is defined for a particular area, value, or formula in a workbook. Use GET.DEF to get the name corresponding to a definition. To get the definition of a name, use GET.NAME.

Syntax

GET.DEF(**def_text**, document_text, type_num)

Def_text can be anything you can define a name to refer to, including a reference, a value, an object, or a formula.

- References must be given in R1C1 style, such as "R3C5".
- If def_text is a value or formula, it is not necessary to include the equal sign that is displayed in the Refers To box in the Define Name dialog box, which appears when you choose the Name command from the Define submenu on the Insert Menu.
- If there is more than one name for def_text, GET.DEF returns the first name. If no name matches def_text, GET.DEF returns the #NAME? error value.

Document_text specifies the sheet or macro sheet that def_text is on. If document_text is omitted, it is assumed to be the active macro sheet.

Type_num is a number from 1 to 3 specifying which types of names are returned.

Type_num	Returns
1 or omitted	Normal names only
2	Hidden names only
3	All names

Examples

If the specified range in Sheet4 is named Sales, the following macro formula returns "Sales":

```
GET.DEF("R2C2:R9C6", "Sheet4")
```

If the value 100 in Sheet4 is defined as Constant, the following macro formula returns "Constant":

```
GET.DEF("100", "Sheet4")
```

If the specified formula in Sheet4 is named SumTotal, the following macro formula returns "SumTotal":

```
GET.DEF("SUM(R1C1:R10C1)", "Sheet4")
```

If 3 is defined as the hidden name Counter on the active macro sheet, the following macro formula returns "Counter":

```
GET.DEF("3", , 2)
```

Related Functions

GET.CELL   Returns information about the specified cell

GET.NAME   Returns the definition of a name

GET.NOTE   Returns characters from a note

NAMES   Returns the names defined on a workbook

GET.DOCUMENT

Returns information about a sheet in a workbook.

Syntax

GET.DOCUMENT(**type_num**, name_text)

Type_num   is a number that specifies what type of information you want. The following lists show the possible values of type_num and the corresponding results.

Type_num	Returns
1	Returns the name of the workbook and worksheet as text. If there is only one sheet in the workbook and the sheet name is the same as the workbook name less any extension, returns the name of the book. The book name does not include the drive, directory or folder, or window number. Otherwise, returns the book and sheet name in the form "[BOOK1.XLS]Sheet1". It is usually best to use GET.DOCUMENT(76) and GET.DOCUMENT(88) to return the name of the active worksheet and the active workbook.
2	Path of the directory or folder containing name_text, as text. If the workbook name_text hasn't been saved yet, returns the #N/A error value.
3	Number indicating the type of sheet. If name_text is a sheet, then the return value is one of the following numbers. If name_text is a book, then the return value is always 5. If name_text is omitted, then the sheet type is returned. If the book has one sheet that is named the same as the book, then the sheet type is returned.

	1 = Worksheet 2 = Chart 3 = Macro sheet 4 = Info window if active 5 = Reserved 6 = Module 7 = Dialog
4	If changes have been made to the sheet since it was last saved, returns TRUE; otherwise, returns FALSE.
5	If the sheet is read-only, returns TRUE; otherwise, returns FALSE.
6	If the sheet is password protected, returns TRUE; otherwise, returns FALSE.
7	If cells in a sheet, the contents of a sheet, or the series in a chart are protected, returns TRUE; otherwise, returns FALSE.
8	If the workbook windows are protected, returns TRUE; otherwise, returns FALSE.

The next four values of type_num apply only to charts.

Type_num	**Returns**
9	Number indicating the type of the main chart: 1 = Area 2 = Bar 3 = Column 4 = Line 5 = Pie 6 = XY (scatter) 7 = 3-D area 8 = 3-D column 9 = 3-D line 10 = 3-D pie

	11 = Radar 12 = 3-D bar 13 = 3-D surface 14 = Doughnut
10	Number indicating the type of the overlay chart. Same as 1, 2, 3, 4, 5, 6, 11, and 14 for main chart above. If there is no overlay chart, returns the #N/A error value.
11	Number of series in the main chart.
12	Number of series in the overlay chart.

The next values of type_num apply to worksheets and macro sheets and to charts when appropriate.

Type_num	**Returns**
9	Number of the first used row. If the sheet is empty, returns 0.
10	Number of the last used row. If the sheet is empty, returns 0.
11	Number of the first used column. If the sheet is empty, returns 0.
12	Number of the last used column. If the sheet is empty, returns 0.
13	Number of windows.
14	Number indicating calculation mode: 1 = Automatic 2 = Automatic except tables 3 = Manual
15	If the Iteration check box is selected in the Calculation tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.

16	Maximum number of iterations.
17	Maximum change between iterations.
18	If the Update Remote References check box is selected in the Calculation tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
19	If the Precision As Displayed check box is selected in the Calculation tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
20	If the 1904 Date System check box is selected in the Calculation tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.

Type_num values of 21 through 29 correspond to the four default fonts in previous versions of Microsoft Excel. These values are provided only for macro compatibility.

The next values of type_num apply to worksheets and macro sheets, and to charts if indicated.

Type_num	**Returns**
30	Horizontal array of consolidation references for the current sheet, in the form of text. If the list is empty, returns the #N/A error value.
31	Number from 1 to 11, indicating the function used in the current consolidation. The function that corresponds to each number is listed under the CONSOLIDATE function. The default function is SUM.
32	Three-item horizontal array indicating the status of the check boxes in the Data Consolidate dialog box. An item is TRUE if the check box is selected or FALSE if the check box is cleared. The first item indicates the Top Row check box, the second the Left Column check box, and the third the Create Links To Source Data check box.
33	If the Recalculate Before Save check box is selected in the Calculation tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.

34	If the workbook is read-only recommended, returns TRUE; otherwise, returns FALSE.
35	If the workbook is write-reserved, returns TRUE; otherwise, returns FALSE.
36	If the workbook has a write-reservation password and it is opened with read/write permission, returns the name of the user who originally saved the file with the write-reservation password. If the file is opened as read-only, or if a password has not been added to the workbook, returns the name of the current user.
37	Number corresponding to the file type of the workbook as displayed in the Save As dialog box. See the SAVE.AS function for a list of all the file types that Microsoft Excel recognizes.
38	If the Summary Rows Below Detail check box is selected in the Outline dialog box, returns TRUE; otherwise, returns FALSE.
39	If the Summary Columns To Right Of Detail check box is selected in the Outline dialog box, returns TRUE; otherwise, returns FALSE.
40	If the Always Create Backup check box is selected in the Save Options dialog box, returns TRUE; otherwise, returns FALSE.
41	Number from 1 to 3 indicating whether objects are displayed: 1 = All objects are displayed 2 = Placeholders for pictures and charts 3 = All objects are hidden
42	Horizontal array of all objects in the sheet. If there are no objects, returns the #N/A error value.

43	If the Save External Link Values check box is selected in the Calculation tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
44	If objects in a workbook are protected, returns TRUE; otherwise, returns FALSE.
45	A number from 0 to 3 indicating how windows are synchronized: 0 = Not synchronized 1 = Synchronized horizontally 2 = Synchronized vertically 3 = Synchronized horizontally and vertically
46	A seven-item horizontal array of print settings that can be set by the LINE.PRINT macro function: Setup text Left margin Right margin Top margin Bottom margin Page length A logical value indicating whether output will be formatted (TRUE) or unformatted (FALSE) when printed
47	If the Transition Formula Evaluation check box is selected in the Transition tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
48	The standard column width setting.

The next values of type_num correspond to printing and page settings.

Type_num	**Returns**
49	The starting page number, or the #N/A error value if none is specified or if "Auto" is entered in the First page Number text box on the Page tab of the Page Setup dialog box.

50	The total number of pages that would be printed based on current settings, excluding notes, or 1 if the document is a chart.
51	The total number of pages that would be printed if you print only notes, or the #N/A error value if the document is a chart.
52	Four-item horizontal array indicating the margin settings (left, right, top, bottom) in the currently specified units.
53	A number indicating the orientation: 1 = Portrait 2 = Landscape
54	The header as a text string, including formatting codes.
55	The footer as a text string, including formatting codes.
56	Horizontal array of two logical values corresponding to horizontal and vertical centering.
57	If row or column headings are to be printed, returns TRUE; otherwise, returns FALSE.
58	If gridlines are to be printed, returns TRUE; otherwise, returns FALSE.
59	If the sheet is printed in black and white only, returns TRUE; otherwise, returns FALSE.
60	A number from 1 to 3 indicating how the chart will be sized when it's printed: 1 = Size on screen 2 = Scale to fit page 3 = Use full page
61	A number indicating the pagination order: 1 = Down, then over 2 = Over, then down Returns the #N/A error value if the document is a chart.

62	Percentage of reduction or enlargement, or 100% if none is specified. Returns the #N/A error value if not supported by the current printer or if the document is a chart.
63	A two-item horizontal array indicating the number of pages to which the printout should be scaled to fit, with the first item equal to the width (or #N/A if no width scaling is specified) and the second item equal to the height (or #N/A if no height scaling is specified). #N/A is also returned if the document is a chart.
64	An array of row numbers corresponding to rows that are immediately below a manual or automatic page break.
65	An array of column numbers corresponding to columns that are immediately to the right of a manual or automatic page break.

Note GET.DOCUMENT(62) and GET.DOCUMENT(63) are mutually exclusive. If one returns a value, then the other returns the #N/A error value.

The next values of type_num correspond to various workbook settings.

Type_num	**Returns**
66	In Microsoft Excel for Windows, if the Transition Formula Entry check box is selected in the Transition tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
67	Microsoft Excel version 5.0 or later always returns TRUE here.
68	Microsoft Excel version 5.0 or later always returns the book name.
69	Returns TRUE if Page Breaks is chosen in the View tab of the Options dialog box; otherwise, returns FALSE.
70	Returns the names of all PivotTable reports in the current sheet as a horizontal array.
71	Returns an horizontal array of all the styles in a workbook.

72	Returns an horizontal array of all chart types displayed on the current sheet.
73	Returns an array of the number of series in each chart of the current sheet.
74	Returns the object ID of the control that currently has the focus on a running user-defined dialog (based on the dialog sheet).
75	Returns the object ID of the object that is the current default button on a running user-defined dialog (based on the dialog sheet).
76	Returns the name of the active sheet or macro sheet in the form [Book1]Sheet1.
77	In Microsoft Excel for Windows, returns the paper size, as integer: 1 = Letter 8.5 x 11 in 2 = Letter Small 8.5 x 11 in 5 = Legal 8.5 x 14 in 9 = A4 210 x 297 mm 10 = A4 Small 210 x 297 mm 13 = B5 182 x 257 mm 18 = Note 8.5 x 11 in
78	Returns the print resolution, as a horizontal array of two numbers.
79	Returns TRUE if the Draft Quality check box has been selected from the sheet tab in the Page Setup dialog box; otherwise, returns FALSE.
80	Returns TRUE if the Comments checkbox has been selected on the Sheet tab in the Page Setup dialog box; otherwise, returns FALSE.
81	Returns the Print Area from the Sheet tab of the Page Setup dialog box as a cell reference.

82	Returns the Print Titles from the Sheet tab of the Page Setup dialog box as an array of cell references.
83	Returns TRUE if the worksheet is protected for scenarios; otherwise, returns FALSE.
84	Returns the value of the first circular reference on the sheet, or #N/A if there are no circular references.
85	Returns the advanced filter mode state of the sheet. This is the mode without drop-down arrows on top. Returns TRUE if the list has been filtered by clicking Filter, then Advanced Filter on the Data menu. Otherwise, returns FALSE.
86	Returns the automatic filter mode state of the sheet. This is the mode with drop-down arrows on top. Returns TRUE if you have chosen Filter, then AutoFilter from the Data menu and the filter drop-down arrows are displayed. Otherwise, returns FALSE.
87	Returns the position number of the sheet. The first sheet is position 1. Hidden sheet are included in the count.
88	Returns the name of the active workbook in the form "Book1".

Name_text is the name of an open workbook. If name_text is omitted, it is assumed to be the active workbook.

Examples

The following macro formula returns TRUE if the contents of the active workbook are protected:

```
GET.DOCUMENT(7)
```

In Microsoft Excel for Windows, the following macro formula returns the number of windows in SALES.XLS:

```
GET.DOCUMENT(13, "SALES.XLS")
```

In Microsoft Excel for the Macintosh, the following macro formula returns 3 if the overlay chart on SALES CHART is a column chart:

```
GET.DOCUMENT(10, "SALES CHART")
```

To find out if SHEET1 is password-protected and if its contents and windows are protected, enter the following formula in a three-cell horizontal array:

```
GET.DOCUMENT({6, 7, 8}, "SHEET1")
```

Related Functions

GET.CELL   Returns information about the specified cell

GET.WINDOW   Returns information about a window

GET.WORKSPACE   Returns information about the workspace

GET.FORMULA

Returns the contents of a cell as they would appear in the formula bar. The contents are given as text, for example, "=2*PI()/360". If the formula contains references, they are returned as R1C1-style references, such as "=RC[1]*(1+R1C1)". Use GET.FORMULA to get a formula from a cell in order to edit its arguments. Use GET.CELL(6) to get a formula in either A1 or R1C1 format, depending on the workspace setting.

Syntax

GET.FORMULA(**reference**)

Reference   is a cell or range of cells on a sheet or macro sheet.

- If a range of cells is selected, GET.FORMULA returns the contents of the upper-left cell in reference.
- Reference can be an external reference.
- Reference can be the object identifier of a picture created by the camera tool.
- Reference can also be a reference to a chart series in the form "Sn" where n is the number of the series. When a chart series is specified, GET.FORMULA returns the series formula using R1C1-style references.

Tip   If you want to get the formula in the active cell, use the ACTIVE.CELL function as the reference argument.

Examples

If cell A3 on the active sheet contains the number 523, then:

`GET.FORMULA(!$A$3)` equals "523"

If cell C2 on the active sheet contains the formula =B2*(1+$A$1), then:

`GET.FORMULA(!$C$2)` equals "=RC[-1]*(1+R1C1)"

The following macro formula returns the contents of the active cell on the active sheet:

```
GET.FORMULA(ACTIVE.CELL())
```

Related Functions

GET.CELL   Returns information about the specified cell

GET.DEF   Returns a name matching a definition

GET.NAME   Returns the definition of a name

GET.NOTE   Returns characters from a comment

GET.LINK.INFO

Returns information about the specified link. Use GET.LINK.INFO to get information about the update settings of a link.

Syntax

GET.LINK.INFO(**link_text, type_num**, type_of_link, reference)

Link_text   is the path of the link as displayed in the Links dialog box, which appears when you choose the Links command from the Edit menu. The path to the file you wish to return DDE information on must be surrounded by single quotes.

Type_num   is a number that specifies what type of information about the currently selected link to return. Type_num 2 applies only to publishers and subscribers in Microsoft Excel for the Macintosh.

Type_num	Returns
1	If the link is set to automatic update, returns 1; otherwise 2.
2	Date of the latest edition as a serial number. Returns #N/A if link_text is not a publisher or a subscriber.

Type_of_link   is a number from 1 to 6 that specifies what type of link you want to get information about.

Type_of_link	Link document type
1	Not applicable
2	DDE link (Microsoft Windows)
3	Not applicable
4	Not applicable
5	Publisher (Macintosh)
6	Subscriber (Macintosh)

Reference   specifies the cell range in R1C1 format of the publisher or subscriber that you want information about. Reference is required if you have more than one publisher or subscriber of a single edition name on the active workbook. Use reference to specify the location of the subscriber you want to return information about. If the subscriber is a picture, or if the publisher is an embedded chart, reference is the number of the object as displayed in the Name box.

Remarks

- If Microsoft Excel cannot find link_text, or if type_of_link does not match the link specified by link_text, GET.LINK.INFO returns the #VALUE! error value.
- If you have more than one subscriber to the edition link_text or if the same area is published more than once, you must specify reference.

Example

In Microsoft Excel for Windows, the following macro formula returns information about a DDE link to a Microsoft Word for Windows document. The document is named NEWPROD.DOC.

```
GET.LINK.INFO("WinWord|'C:\WINWORD\NEWPROD.DOC'!DDE_LINK1", 1, 2)
```

In Microsoft Excel for the Macintosh, the following macro formula returns information about a link to a publisher defined in cells A1:C3 on a workbook named New Products.

```
GET.LINK.INFO("A1:C3 New Products Edition #1", 2, 5, "'New
Products'!R1C1:R3C3")
```

Related Functions

CREATE.PUBLISHER   Creates a publisher from the selection

SUBSCRIBE.TO   Inserts contents of an edition into the active workbook

UPDATE.LINK   Updates a link to another workbook

GET.NAME

Returns the definition of a name as it appears in the Refers To box of the Define Name dialog box, which appears when you choose the Define command from the Name submenu on the Insert menu. If the definition contains references, they are given as R1C1-style references. Use GET.NAME to check the value defined by a name. To get the name corresponding to a definition, use GET.DEF.

Syntax

GET.NAME(**name_text**, info_type)

Name_text   can be a name defined on the macro sheet; an external reference to a name defined on the active workbook, for example, "!Sales"; or an external reference to a name defined on a particular open workbook, for example, "[Book1]SHEET1!Sales". Name_text can also be a hidden name.

Info_type   specifies the type of information to return about the name. If 1 or omitted, the definition is returned. If 2, returns TRUE if the name is defined for just the sheet, FALSE if the name is defined for the entire workbook.

Remarks

If the Contents check box has been selected in the Protect Sheet dialog box to protect the workbook containing the name, GET.NAME returns the #N/A error value. To see the Protect Sheet dialog box, choose the Protect Sheet command on the Protection submenu from the Tools menu.

Examples

If the name Sales on a macro sheet is defined as the number 523, then:

`GET.NAME("Sales")` equals "=523"

If the name Profit on the active sheet is defined as the formula =Sales-Costs, then:

`GET.NAME("!Profit")` equals "=Sales-Costs"

If the name Database on the active sheet is defined as the range A1:F500, then:

`GET.NAME("!Database")` equals "=R1C1:R500C6"

Related Functions

DEFINE.NAME	Defines a name on the active or macro sheet
GET.CELL	Returns information about the specified cell
GET.DEF	Returns a name matching a definition
NAMES	Returns the names defined in a workbook
SET.NAME	Defines a name as a value

GET.NOTE

Returns characters from a comment.

Syntax

GET.NOTE(cell_ref, start_char, num_chars)

Cell_ref is the cell to which the note is attached. If cell_ref is omitted, the comment attached to the active cell is returned.

Start_char is the number of the first character in the comment to return. If start_char is omitted, it is assumed to be 1, the first character in the comment.

Num_chars is the number of characters to return. Num_chars must be less than or equal to 255. If num_chars is omitted, it is assumed to be the length of the comment attached to cell_ref.

Examples

The following macro formula returns the first 200 characters in the comment attached to cell A3 on the active sheet:

```
GET.NOTE(!$A$3, 1, 200)
```

In Microsoft Excel for Windows, the following macro formula returns the 10th through the 39th characters of the comment attached to cell C2 on SALES.XLS:

```
GET.NOTE("[SALES.XLS]Sheet1!R2C3", 10, 30)
```

In Microsoft Excel for the Macintosh, the following macro formula returns the 10th through the 39th characters of the comment attached to cell C2 on SALES:

```
GET.NOTE("[SALES]Sheet1!R2C3", 10, 30)
```

Use GET.NOTE with the NOTE function to move the contents of a comment to a cell or text box or to another comment attached to a cell:

```
NOTE(GET.NOTE(!$B$10),ACTIVE.CELL())
```

Related Functions

GET.CELL   Returns information about the specified cell

NOTE   Creates or changes a comment.

GET.OBJECT

Returns information about the specified object. Use GET.OBJECT to return information you can use in other macro formulas that manipulate objects.

Syntax

GET.OBJECT(**type_num**, object_id_text, start_num, count_num, item_index)

Type_num   is a number specifying the type of information you want returned about an object. GET.OBJECT returns the #VALUE! error value (and the macro is halted) if an object isn't specified or if more than one object is selected.

Type_num	**Returns**
2	If the object is locked, returns TRUE; otherwise FALSE.
3	Z-order position (layering) of the object; that is, the relative position of the overlapping objects, starting with 1 for the object that is most under the others.
4	Reference of the cell under the upper-left corner of the object as text in R1C1 reference style; for a line or arc, returns the start point.
5	X offset from the upper-left corner of the cell under the upper-left corner of the object, measured in points.
6	Y offset from the upper-left corner of the cell under the upper-left corner of the object, measured in points.

7	Reference of the cell under the lower-right corner of the object as text in R1C1 reference style; for a line or arc, returns the end point.
8	X offset from the upper-left corner of the cell under the lower-right corner of the object, measured in points.
9	Y offset from the upper-left corner of the cell under the lower-right corner of the object, measured in points.
10	Name, including the filename, of the macro assigned to the object. If no macro is assigned, returns FALSE.
11	Number indicating how the object moves and sizes: 1 = Object moves and sizes with cells 2 = Object moves with cells 3 = Object is fixed

Values 12 to 21 for type_num apply only to text boxes and buttons. If another type of object is selected, GET.OBJECT returns the #VALUE! error value.

Type_num	**Returns**
12	Text starting at start_num for count_num characters.
13	Font name of all text starting at start_num for count_num characters. If the text contains more than one font name, returns the #N/A error value.
14	Font size of all text starting at start_num for count_num characters. If the text contains more than one font size, returns the #N/A error value.
15	If all text starting at start_num for count_num characters is bold, returns TRUE. If text contains only partial bold formatting, returns the #N/A error value.
16	If all text starting at start_num for count_num characters is italic, returns TRUE. If text contains only partial italic formatting, returns the #N/A error value.

17	If all text starting at start_num for count_num characters is underlined, returns TRUE. If text contains only partial underline formatting, returns the #N/A error value.
18	If all text starting at start_num for count_num characters is struck through, returns TRUE. If text contains only partial struck-through formatting, returns the #N/A error value.
19	In Microsoft Excel for the Macintosh, if all text starting at start_num for count_num characters is outlined, returns TRUE. If text contains only partial outline formatting, returns the #N/A error value. Always returns FALSE in Microsoft Excel for Windows.
20	In Microsoft Excel for the Macintosh, if all text starting at start_num for count_num characters is shadowed, returns TRUE. If text contains only partial shadow formatting, returns the #N/A error value. Always returns FALSE in Microsoft Excel for Windows.
21	Number from 0 to 56 indicating the color of all text starting at start_num for count_num characters; if color is automatic, returns 0. If more than one color is used, returns the #N/A error value.

Values 22 to 25 for type_num also apply only to text boxes and buttons. If another type of object is selected, GET.OBJECT returns the #N/A error value.

Type_num	**Returns**
23	Number indicating the vertical alignment of text: 1 = Top 2 = Center 3 = Bottom 4 = Justified
24	Number indicating the orientation of text: 0 = Horizontal 1 = Vertical 2 = Upward 3 = Downward
25	If button or text box is set to automatic sizing, returns TRUE; otherwise FALSE.

The following values for type_num apply to all objects, except where indicated.

Type_num	Returns
27	Number indicating the type of the border or line: 0 = Custom 1 = Automatic 2 = None
28	Number indicating the style of the border or line as shown in the Patterns tab in the Format Objects dialog box: 0 = None 1 = Solid line 2 = Dashed line 3 = Dotted line 4 = Dashed dotted line 5 = Dashed double-dotted line 6 = 50% gray line 7 = 75% gray line 8 = 25% gray line
29	Number from 0 to 56 indicating the color of the border or line; if the border is automatic, returns 0.
30	Number indicating the weight of the border or line: 1 = Hairline 2 = Thin 3 = Medium 4 = Thick
31	Number indicating the type of fill: 0 = Custom 1 = Automatic 2 = None
32	Number from 1 to 18 indicating the fill pattern as shown in the Format Object dialog box.
33	Number from 0 to 56 indicating the foreground color of the fill pattern; if the fill is automatic, returns 0. If the object is a line, returns the #N/A error value.

34 Number from 0 to 56 indicating the background color of the fill pattern; if the fill is automatic, returns 0. If the object is a line, returns the #N/A error value.

35 Number indicating the width of the arrowhead:
1 = Narrow
2 = Medium
3 = Wide
If the object is not a line, returns the #N/A error value.

36 Number indicating the length of the arrowhead:
1 = Short
2 = Medium
3 = Long
If the object is not a line, returns the #N/A error value.

37 Number indicating the style of the arrowhead:
1 = No head
2 = Open head
3 = Closed head
4 = Open double-ended head
5 = Closed double-ended head
If the object is not a line, returns the #N/A error value.

38 If the border has round corners, returns TRUE; if the corners are square, returns FALSE. If the object is a line, returns the #N/A error value.

39 If the border has a shadow, returns TRUE; if the border has no shadow, returns FALSE. If the object is a line, returns the #N/A error value.

40 If the Lock Text check box in the Protection Tab of the Format Object dialog box is selected, returns TRUE; otherwise FALSE.

41 If objects are set to be printed, returns TRUE; otherwise FALSE.

42 The horizontal distance, measured in points, from the left edge of the active window to the left edge of the object. May be a negative number if the window is scrolled beyond the object.

43	The vertical distance, measured in points, from the top edge of the active window to the top edge of the object. May be a negative number if the window is scrolled beyond the object.
44	The horizontal distance, measured in points, from the left edge of the active window to the right edge of the object. May be a negative number if the window is scrolled beyond the object.
45	The vertical distance, measured in points, from the top edge of the active window to the bottom edge of the object. May be a negative number if the window is scrolled beyond the object.
46	The number of vertices in a polygon, or the #N/A error value if the object is not a polygon.
47	A count_num by 2 array of vertex coordinates starting at start_num in a polygon's array of vertices.
48	If the object is a text box, returns the cell reference that the text box is linked to. If the object is a control on a worksheet, returns the cell reference that the control's value is linked to. This information is returned as a string.
49	Returns the ID number of the object. For example, "Rectangle 5" returns 5. Note that the name of the object may not have this index in it if the object has been renamed by the user.
50	Returns the object's classname. For example, "Rectangle".
51	Returns the object name. By default, object names are the classname followed by the ID. For example, "Rectangle 1" is an object name, of which "Rectangle" is the classname, and 1 is the ID number. The object can also be renamed, in which case the name picked by the user is returned.
52	Returns the distance from cell A1 to the Left of the object bounding rectangle in points
53	Returns the distance from Cell A1 to the top of the object bounding rectangle in points

54	Returns the width of object bounding rectangle in points
55	Returns the height of object bounding rectangle in points
56	If the object is enabled, returns TRUE; otherwise, it returns FALSE.
57	Returns the shortcut key assignment for the control object, as text.
58	Returns TRUE is the button control on a dialog sheet is the default button of the dialog; otherwise, returns FALSE
59	Returns TRUE if the button control on the dialog sheet is clicked when the user presses the ESCAPE Key; otherwise, returns FALSE.
60	Returns TRUE if the button control on a dialog sheet will close the dialog box when pressed; otherwise, returns FALSE
61	Returns TRUE if the button control on a dialog sheet will be clicked when the user presses F1.
62	Returns the value of the control. For a check box or radio button, Returns 1 if it is selected, zero if it is not selected, or 2 if mixed. For a List box or dropdown box, returns the index number of the selected item, or zero if no item is selected. For a scroll bar, returns the numeric value of the scroll bar.
63	Returns the minimum value that a scroll bar or spinner button can have
64	Returns the maximum value that a scroll bar or spinner button can have
65	Returns the step increment value added or subtracted from the value of a scroll bar or spinner. This value is used when the arrow buttons are pressed on the control.
66	Returns the large, or "page" step increment value added or subtracted from the value of a scroll bar when it is clicked in the region between the thumb and the arrow buttons.

67 Returns the input type allowed in an edit box control:
1 = Text
2 = Integer
3 = Number (what type)
4 = Cell reference
5 = Formula

68 Returns TRUE if the edit box control allows multi-line editing with wrapped text; otherwise, it returns FALSE.

69 Returns TRUE if the edit box has a vertical scroll bar; otherwise, it returns FALSE.

70 Returns the object ID of the object that is linked to a list box or edit box. For a dropdown combo box that has an editable entry field, returns the object ID of itself. A dropdown box that can't be edited, returns FALSE.

71 Returns the number of entries in a List box, dropdown List box, or dropdown combo box.

72 Returns the text of the selected entry in a List box, dropdown List box, or dropdown combo box.

73 Returns the range used to fill the entries in a List box, dropdown List box, or dropdown combo box, as text. If an empty string is returned, then the control isn't filled from a range.

74 Returns the number of list lines displayed when a dropdown control is dropped.

75 Returns TRUE the object is displayed as 3-D; otherwise, it returns FALSE.

76 Returns the Far East phonetic accelerator key as text. Used for Far East versions of Microsoft Excel.

77 Returns the select status of the list box:
0 = single
1 = simple multi-select
2 = extended multi-select

78	Returns an array of TRUE and FALSE values indicating which items are selected in a list box. If TRUE, the item is selected; If FALSE, the item is not selected.
79	Returns TRUE if the add indent attribute is on for alignment. Returns FALSE if the add indent attribute is off for alignment. Used for only Far East versions of Microsoft Excel.

Object_id_text is the name and number, or number alone, of the object you want information about. Object_id_text is the text displayed in the reference area when the object is selected. If object_id_text is omitted, it is assumed to be the selected object. If object_id_text is omitted and no object is selected, GET.OBJECT returns the #REF! error value and interrupts the macro.

Start_num is the number of the first character in the text box or button or the first vertex in a polygon you want information about. Start_num is ignored unless a text box, button, or polygon is specified by type_num and object_id_text. If start_num is omitted, it is assumed to be 1.

Count_num is the number of characters in a text box or button, or the number of vertices in a polygon, starting at start_num, that you want information about. Count_num is ignored unless a text box, button, or polygon is specified by type_num and object_id_text. If count_num is omitted, it is assumed to be 255.

Item_index is the index number or position of the item in the list box or drop-down box that you want information about, ranging from 1 to the number of items in the list box or drop-down box.

Tip Use GET.OBJECT(45) - GET.OBJECT(43) to determine the height of an object and GET.OBJECT(44) - GET.OBJECT(42) to determine the width.

Examples

The following macro formula returns the reference of the cell under the upper-left corner of the object Oval 3 (assume the cell is E2):

`GET.OBJECT(4, "Oval 3")` returns "R2C5"

The following macro formula changes the protection status of the object Rectangle 2 if it is locked:

```
IF(GET.OBJECT(2, "Rectangle 2"), OBJECT.PROTECTION(FALSE))
```

The following macro formula returns characters 25 through 185 from the object Text 5:

```
GET.OBJECT(12, "Text 5", 25, 160)
```

Related Functions

CREATE.OBJECT	Creates an object
FONT.PROPERTIES	Applies a font to the selection
OBJECT.PROTECTION	Controls how an object is protected
PLACEMENT	Determines an object's relationship to underlying cells

GET.PIVOT.FIELD

Returns information about a field in a PivotTable report.

Syntax

GET.PIVOT.FIELD(type_num, pivot_field_name, pivot_table_name)

Type_num   is a value from 1 to 17 that returns the following types of information:

Type_num	**Value**
1	Returns an array of all the items which make up pivot_field_name. The array is made up of text constants, dates or numbers depending on the field.
2	Returns an array of all items which are set to show with the pivot_field_name. The array is made up of text constants, dates or numbers depending on the field. The array is returned in the order that the items are displayed in the PivotTable report. If pivot_field_name is a page field, then the array contains only one element, the value corresponding to the active page (this could be all if the All item is showing).
3	Returns an array of all items which are hidden in the pivot_field_name. The array is made up of text constants, dates or numbers depending on the field. If pivot_field_name is a data field or the data header name, this function returns the #N/A! error value.
4	Returns an integer describing where the field is displayed in the active PivotTable report (either row or column): 0 = Hidden 1 = Row 2 = Col 3 = Page 4 = Data
5	Returns an array of all items in pivot_field_name that are group parents. The array is made up of text constants, dates or numbers depending on the field. The array is returned in the order which these items appear in the PivotTable report. Returns #N/A if there are no group parents and if the pivot_field_name is a data field or the data field header.

6 Returns a number between 0 and 4095 which describes the subtotals attached to the field. The number is the sum of the values associated with each subtotal function. See PIVOT.FIELD.PROPERTIES for a list of all the values associated with subtotal calculations. If the field is showing as a data field or data field header, #N/A! is returned.

7 Returns an integer describing the type of data contained in the field:

0 = Text

1 = Number

2 = Date

8 Returns an array five columns wide and one row high describing the summary function's custom calculation shown with the specified field (Data field) in the PivotTable report. The array will look as follows: {function, calculation, base field, base item, number format}. If pivot_field_name is not showing in the active PivotTable report as a data field, #N/A! is returned.

9 Returns a reference to all of pivot_field_name's items currently showing in the active PivotTable report. If pivot_field_name is hidden, #N/A! is returned. If pivot_field_name is a page field, the reference to the currently showing page item is returned. If pivot_field_name is a data field, a reference to all the data for this field in the PivotTable report is returned. The references are returned as text.

10 Returns a reference to the header cell for pivot_field_name. If pivot_field_name is a data field, a reference to all the headers in the data row or column is returned. If pivot_field_name is hidden, #N/A! is returned. The reference is returned as text.

11 Returns the number of grouped fields in the grouped field set which includes pivot_field_name. If pivot_field_name is neither a parent field nor a child field, 1 is returned. If pivot_field_name is a data field or data header name, the function returns the #N/A! error value.

12 Returns the level of pivot_field_name in the grouped field set which includes pivot_field_name. Returns 1 for the highest level parent field, 2 for its child field, and so on. If pivot_field_name is neither a parent field nor a child field, 1 is returned. If pivot_field_name is a data field or data header name, the function returns the #N/A! error value.

13 Returns the name of the parent field for pivot_field_name as a text constant. If pivot_field_name is not a child field, #N/A! is returned.

14 Returns the name of the child field for pivot_field_name as a text constant. If pivot_field_name is not a parent field, #N/A! is returned.

15	Returns a text constant representing the original name of the field in the data source.
16	Returns the position of the field among all the other fields in its orientation. For instance, a 1 would be returned if the field was the first row field.
17	Returns an array of all items in pivot_field_name that are group children. The array is made up of text constants, dates or numbers depending on the field. The array is returned in the order which these items appear in the PivotTable report. Returns #N/A if there are no group children, and if the pivot_field_name is a data field or the data field header.

Pivot_field_name is the name of the field that you want information about. If there is no field named pivot_field_name in the PivotTable report, returns #VALUE!.

Pivot_table_name is the name of a PivotTable report containing the field that you want information about. If omitted, the PivotTable report containing the active cell is used. If the active cell is not in a PivotTable report, the #VALUE! error value is returned.

Related Functions

GET.PIVOT.ITEM Returns information about an item in a PivotTable report.

GET.PIVOT.TABLE Returns information about a PivotTable report.

GET.PIVOT.ITEM

Returns information about an item in a PivotTable report.

Syntax

GET.PIVOT.ITEM(**type_num**, pivot_item_name, pivot_field_name, pivot_table_name)

Type_num is a value from 1 to 9 the represents the type of information you want about an item in a PivotTable report.

Type_num	**Information**
1	Returns the position of the item in its field. Returns #N/A if pivot_field_name is a data field. Returns #N/A! if the item is hidden.
2	Returns the reference to all the cells in the PivotTable header currently containing pivot_item_name. This reference is returned as text. If pivot_item_name is currently not showing in the PivotTable report, #N/A! is returned.

3	Returns the reference to all the data in the PivotTable report which is qualified by pivot_item_name. This reference is returned as text. If pivot_item_name is currently not showing in the PivotTable report, #N/A! is returned.
4	Returns an array of text constants representing the children of pivot_item_name if pivot_item_name is a parent. Otherwise the function returns #N/A!.
5	Returns a text constant representing the parent of pivot_item_name, if pivot_item_name exists as part of a group. Otherwise the function returns #N/A!.
6	Returns TRUE if pivot_item_name is a member of a group which is currently expanded to show detail. Returns FALSE if pivot_item_name is a member of a group currently collapsed to hide detail. If pivot_item_name is not a member of a group, the function returns #N/A!.
7	Returns TRUE if pivot_item_name is expanded to show detail. Returns FALSE if pivot_item_name is collapsed to hide detail.
8	Returns TRUE if the item pivot_item_name is currently visible, FALSE if it is hidden.
9	Returns the name of the item as it appeared in the original at a source. This will differ from the current item name only if the user changes the name of the item after creating the PivotTable report.

Pivot_item_name is the name of the item that you want information about. If there is no item named pivot_item_name in the PivotTable report, returns #VALUE!.

Pivot_field_name is the name of the field that you want information about. If there is no field named pivot_field_name in the PivotTable report, returns #VALUE!.

Pivot_table_name is the name of a PivotTable report containing the field that you want information about. If omitted, uses the PivotTable report containing the active cell. If the active cell is not in a PivotTable report, the #VALUE! error value is returned.

Related Functions

GET.PIVOT.FIELD Returns information about an item in a PivotTable report.

GET.PIVOT.TABLE Returns information about a PivotTable report.

GET.PIVOT.TABLE

Returns information about a PivotTable report.

Syntax

GET.PIVOT.TABLE(**type_num**,pivot_table_name)

Type_num is a value from 1 to 22 that represents a type of information you want about a PivotTable report.

Type_num	**Information**
1	Returns the name of the person who last updated the PivotTable report, as a text constant.
2	Returns the date the PivotTable report was last updated, as a serial number.
3	Returns a horizontal array of text constants representing all the fields in the PivotTable report.
4	Returns an integer representing the number of fields in the PivotTable report.
5	Returns a horizontal array of text constants representing all the visible fields in the PivotTable report (rows, columns, pages or data)
6	Returns a horizontal array of text constants representing all the hidden fields in the PivotTable report. Return #N/A if no hidden fields.
7	Returns a horizontal array of text constants representing the names of all the fields currently showing in the PivotTable report as row fields. Returns #N/A if there are no row fields.
8	Returns a horizontal array of text constants representing all the fields currently showing in the PivotTable report as column fields. Returns #N/A if no column fields exist.

9	Returns a horizontal array of text constants representing all the fields currently showing in the PivotTable report as page fields. Return #N/A if no page fields exist.
10	Returns a horizontal array of text constants representing all the fields currently showing in the PivotTable report as data fields. Returns #N/A if there are no data fields.
11	Returns the smallest rectangular reference which bounds the PivotTable report and all headers (not including the page header). This reference is returned as text.
12	Returns the smallest rectangular reference which bounds the PivotTable report and all headers (including the page headers). This reference is returned as text.
13	Returns the reference to the row header area as text. The row header area includes each row field header along with all the items in each row field. Returns #N/A if there are no row headers.
14	Returns the reference to the column header area as text. The column header area includes each column field header along with all the items in each column field. Returns #N/A if there are no column headers.
15	Returns the reference to the data header area as text. The data header area includes the data field header along with all the headers in the data row/col. Returns #N/A if there is no data field.
16	Returns a reference to all the page headers as text.
17	Returns the reference to the PivotTable report data area as text.
18	Returns TRUE if the PivotTable report is set to show row grand totals.
19	Returns TRUE if the PivotTable report is set to show column grand totals.

20	Returns TRUE if the user is saving data with the PivotTable report.
21	Returns TRUE if the PivotTable report is set up to Autoformat on pivoting.
22	Returns the data source of the PivotTable report. The kind of information returned depends on the data source: If the data source is a Microsoft Excel list or database, the cell reference is returned as text. If the data source is an external data source, then an array is returned. Each row consists of a SQL connection string with the remaining elements as the query string broken down into 200 character segments. If the data source is Multiple Consolidation ranges, then a two dimensional array is returned, each row of which consists of a reference and associated page field items. If the data source is another PivotTable report, then one of the above three kinds of information is returned.

Pivot_table_name is the name of a PivotTable report containing the field that you want information about. If omitted, uses the PivotTable report containing the active cell.

Remarks

Returns #VALUE! error value when pivot_table_name is not a valid PivotTable name on the active sheet and the active cell is not within a PivotTable report.

Related Functions

GET.PIVOT.FIELD Returns information about an item in a PivotTable report.

GET.PIVOT.ITEM Returns information about a PivotTable report.

GET.TOOL

Returns information about a button or buttons on a toolbar. Use GET.TOOL to get information about a button to use with functions that add, delete, or alter buttons.

Syntax

GET.TOOL(**type_num, bar_id, position**)

Type_num specifies what type of information you want GET.TOOL to return.

Type_num	Returns
1	The button's ID number. Gaps are represented by zeros.
2	The reference of the macro assigned to the button. If no macro is assigned, GET.TOOL returns the #N/A error value.
3	If the button is down, returns TRUE. If the button is up, returns FALSE.
4	If the button is enabled, returns TRUE. If the button is disabled, returns FALSE.
5	A logical value indicating the type of the face on the button:
	TRUE = bitmap
	FALSE = a default button face
6	The help_text reference associated with the custom button. If the button is built-in, returns #N/A.
7	The balloon_text reference associated with the custom button. If the button is built-in, returns the #N/A error value.
8	The Help context string associated with the custom button.
9	The Tip_text associated with the custom button.

Bar_id specifies the number or name of the toolbar for which you want information. For detailed information about bar_id, see ADD.TOOL.

Position specifies the position of the button on the toolbar. Position starts with 1 at the left side (if horizontal) or at the top (if vertical). A position can be occupied by a button or a gap.

Example

The following macro formula requests the help text associated with the third button in Toolbar2:

```
GET.TOOL(6, "Toolbar2", 3)
```

Related Functions

ADD.TOOL   Adds one or more buttons to a toolbar

DELETE.TOOL   Deletes a button from a toolbar

ENABLE.TOOL   Enables or disables a button on a toolbar

GET.TOOLBAR   Retrieves information about a toolbar

GET.TOOLBAR

Returns information about one toolbar or all toolbars. Use GET.TOOLBAR to get information about a toolbar to use with functions that add, delete, or alter toolbars.

Syntax

GET.TOOLBAR(**type_num**, bar_id)

Type_num   specifies what type of information to return. If type_num is 8 or 9, GET.TOOLBAR returns an array of names or numbers of all visible or hidden toolbars. Otherwise, bar_id is required, and GET.TOOLBAR returns the requested information about the specified toolbar.

Type_num	**Returns**
1	A horizontal array of all tool IDs on the toolbar, ordered by position. Gaps are represented by zeros.
2	Number indicating the horizontal position (x-coordinate) of the toolbar in the docked or floating region. For more information, see SHOW.TOOLBAR.
3	Number indicating the vertical position (y-coordinate) of the toolbar in the docked or floating region.
4	Number indicating the width of the toolbar in points.
5	Number indicating the height of the toolbar in points.

6	Number indicating the toolbar location: 1 = Top dock in the workspace 2 = Left dock in the workspace 3 = Right dock in the workspace 4 = Bottom dock in the workspace 5 = Floating
7	If the toolbar is visible, returns TRUE. If the toolbar is hidden, returns FALSE.
8	An array of toolbar IDs (names or numbers in the bar_id array) for all toolbars, visible and hidden.
9	An array of toolbar IDs (names or numbers in the bar_id array) for all visible toolbars.
10	If the toolbar is visible in full-screen mode, returns TRUE; otherwise, returns FALSE.

Bar_id specifies the number or name of a toolbar for which you want information. If type_num is 8 or 9, Microsoft Excel ignores bar_id. For detailed information about bar_id, see ADD.TOOL.

Remarks

If you request position information for a hidden toolbar, Microsoft Excel returns the position where the toolbar would appear if shown.

Examples

The following macro formula returns information about the width of Toolbar1:

```
GET.TOOLBAR(4, "Toolbar1")
```

When the following macro formula is entered as an array with CTRL+SHIFT+ENTER, the IDs of all visible toolbars are returned, and the array is named All_Bar_Ids:

```
SET.NAME("All_Bar_Ids", GET.TOOLBAR(9))
```

Related Functions

ADD.TOOL   Adds one or more buttons to a toolbar

ADD.TOOLBAR   Creates a new toolbar with the specified tools

DELETE.TOOLBAR   Deletes custom toolbars

GET.TOOL   Returns information about a tool or tools on a toolbar

SHOW.TOOLBAR   Hides or displays a toolbar

GET.WINDOW

Returns information about a window. Use GET.WINDOW in a macro that requires the status of a window, such as its name, size, position, and display options.

Syntax

GET.WINDOW(**type_num**, window_text)

Type_num   is a number that specifies what type of window information you want. The following list shows the possible values of type_num and the corresponding results:

Type_num	**Returns**
1	Name of the workbook and sheet in the window as text. For compatibility with Microsoft Excel version 4.0, if the window contains only a single sheet that has the same name as the workbook without its extension, returns only the name of the book. Otherwise, returns the name of the sheet in the form "[Book1]Sheet1".
2	Number of the window.
3	X position, measured in points from the left edge of the workspace (in Microsoft Excel for Windows) or screen (in Microsoft Excel for the Macintosh) to the left edge of the window.
4	Y position, measured in points from the bottom edge of the formula bar to the top edge of the window.
5	Width, measured in points.
6	Height, measured in points.
7	If window is hidden, returns TRUE; otherwise, returns FALSE.

The rest of the values for type_num apply only to worksheets and macro sheets, except where indicated:

Type_num	Returns
8	If formulas are displayed, returns TRUE; otherwise, returns FALSE.
9	If gridlines are displayed, returns TRUE; otherwise, returns FALSE.
10	If row and column headings are displayed, returns TRUE; otherwise, returns FALSE.
11	If zeros are displayed, returns TRUE; otherwise, returns FALSE.
12	Gridline and heading color as a number in the range 1 to 56, corresponding to the colors in the View tab of the Options dialog box; if color is automatic, returns 0.

Values 13 to 16 for type_num return arrays that specify which rows or columns are at the top and left edges of the panes in the window and the widths and heights of those panes.

- The first number in the array corresponds to the first pane, the second number to the second pane, and so on.
- If the edge of the pane occurs at the boundary between rows or columns, the number returned is an integer.
- If the edge of the pane occurs within a row or column, the number returned has a fractional part that represents the fraction of the row or column visible within the pane.
- The numbers can be used as arguments to the SPLIT function to split a window at specific locations.

Type_num	Returns
13	Leftmost column number of each pane, in a horizontal numeric array
14	Top row number of each pane, in a horizontal numeric array.

15	Number of columns in each pane, in a horizontal numeric array.
16	Number of rows in each pane, in a horizontal numeric array.
17	Number indicating the active pane: 1 = Upper, left, or upper-left 2 = Right or upper-right 3 = Lower or lower-left 4 = Lower-right
18	If window has a vertical split, returns TRUE; otherwise, returns FALSE.
19	If window has a horizontal split, returns TRUE; otherwise, returns FALSE.
20	If window is maximized, returns TRUE; otherwise, returns FALSE.
21	Reserved
22	If the Outline Symbols check box is selected in the View tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
23	Number indicating the size of the window (including charts): 1 = Restored 2 = Minimized (displayed as an icon) 3 = Maximized
24	If panes are frozen on the active window, returns TRUE; otherwise, returns FALSE.
25	The numeric magnification of the active window (as a percentage of normal size) as set in the Zoom dialog box, or 100 if none is specified.
26	Returns TRUE if horizontal scrollbars are displayed in the active window; otherwise, returns FALSE.

27	Returns TRUE if vertical scrollbars are displayed in the active window; otherwise, returns FALSE.
28	Returns the tab ratio of workbook tabs to horizontal scrollbar, from 0 to 1. The default is .6.
29	Returns TRUE if workbook tabs are displayed in the active window; otherwise, returns FALSE.
30	Returns the title of the active sheet in the window in the form "[Book1]Sheet1".
31	Returns the name of a workbook only, without read/write indicated. For example, if Book1.xls is read only, then "Book.xls" will be returned without "[Read Only]" appended.

Window_text is the name that appears in the title bar of the window that you want information about. If window_text is omitted, it is assumed to be the active window.

Examples

If the active window contains the workbook Book1, then:

`GET.WINDOW(1)` equals "Book1"

If the title of the active window is Macro1:3, then:

`GET.WINDOW(2)` equals 3

In Microsoft Excel for Windows, the following macro formula returns the gridline and heading color of REPORT.XLS:

```
GET.WINDOW(12, "REPORT.XLS")
```

In Microsoft Excel for the Macintosh, the following macro formula returns the gridline and heading color of REPORT MASTER:

```
GET.WINDOW(12, "REPORT MASTER")
```

Related Functions

GET.DOCUMENT	Returns information about a workbook
GET.WORKSPACE	Returns information about the workspace

GET.WORKBOOK

Returns information about a workbook.

Syntax

GET.WORKBOOK(**type_num**, name_text)

Type_num is a number that specifies what type of workbook information you want.

Type_num	**Returns**
1	The names of all sheets in the workbook, as a horizontal array of text values. Names are returned as [book]sheet.
2	This will always return the #N/A error value.
3	The names of the currently selected sheets in the workbook, as a horizontal array of text values.
4	The number of sheets in the workbook.
5	TRUE if the workbook has a routing slip; otherwise, FALSE.
6	The names of all of the workbook routing recipients who have not received the workbook, as a horizontal array of text values.
7	The subject line for the current routing slip, as text.
8	The message text for the routing slip, as text.
9	If the workbook is to be routed to recipients one after another, returns 1. If it is to be routed all at once, returns 2.
10	TRUE, if the Return When Done check box in the Routing Slip dialog box is selected; otherwise, FALSE.
11	TRUE, if the current recipient has already forwarded the current workbook; otherwise, FALSE.
12	TRUE, if the Track Status checkbox in the Routing Slip dialog box is selected; otherwise, FALSE.
13	Status of the workbook routing slip: 0 = Unrouted

	1 = Routing in progress, or the workbook has been routed to a user 2 = Routing is finished
14	TRUE, if the workbook structure is protected; otherwise, FALSE.
15	TRUE, if the workbook windows are protected; otherwise, FALSE.
16	Name of the workbook as text. The workbook name does not include the drive, directory or folder, or window number.
17	TRUE if the workbook is read only; otherwise, FALSE. This is the equivalent of GET.DOCUMENT(34).
18	TRUE if sheet is write-reserved; otherwise, FALSE. This is the equivalent of GET.DOCUMENT(35).
19	Name of the user with current write permission for the workbook. This is the equivalent of GET.DOCUMENT(36).
20	Number corresponding to the file type of the document as displayed in the Save As dialog box. This is the equivalent of GET.DOCUMENT(37).
21	TRUE if the Always Create Backup check box is selected in the Save Options dialog box; otherwise, FALSE. This is the equivalent of GET.DOCUMENT(40).
22	TRUE if the Save External Link Values check box is selected in the Calculation tab of the Options dialog box. This is the equivalent of GET.DOCUMENT(43).
23	TRUE if the workbook has a PowerTalk mailer; otherwise, FALSE. Returns #N/A if no OCE mailer is installed.
24	TRUE if changes have been made to the workbook since the last time it was saved; FALSE if book is unchanged (or when closed, will not prompt to be saved).
25	The recipients on the To line of a PowerTalk mailer, as a horizontal array of text.
26	The recipients on the Cc line of a PowerTalk mailer, as a horizontal array of text.
27	The recipients on the Bcc line of a PowerTalk mailer, as a horizontal array of text.

28	The subject of the PowerTalk mailer, as text.
29	The enclosures of the PowerTalk mailer, as a horizontal array of text.
30	TRUE, if the PowerTalk mailer has been received from another user (as opposed to just being added but not sent). FALSE, if the mailer has not been received from another user.
31	The date and time the PowerTalk mailer was sent, as a serial number. Returns the #N/A error value if the mailer has not yet been sent.
32	The sender name of the PowerTalk mailer, as text. Returns the #N/A error value if the mailer has not yet been sent.
33	The title of the workbook as displayed on the Summary tab of the Properties dialog box, as text.
34	The subject of the workbook as displayed on the Summary tab of the Properties dialog box, as text.
35	The author of the workbook as displayed on the Summary tab of the Properties dialog box, as text.
36	The keywords for the workbook as displayed on the Summary tab of the Properties dialog box, as text.
37	The comments for the workbook as displayed on the Summary tab of the Properties dialog box, as text.
38	The name of the active sheet.

Name_text is the name of an open workbook. If name_text is omitted, it is assumed to be the active workbook.

Example

The following macro formula returns the name of the active sheet in the workbook named SALES.XLS:

```
GET.WORKBOOK(38, "SALES.XLS")
```

Related Functions

GET.DOCUMENT Returns information about a workbook

WORKBOOK.SELECT Selects the specified documents in a workbook

GET.WORKSPACE

Returns information about the workspace. Use GET.WORKSPACE in a macro that depends on the status of the workspace, such as the environment, version number, and available memory.

Syntax

GET.WORKSPACE(**type_num**)

Type_num is a number specifying the type of workspace information you want. The following list shows the type_num values and their corresponding results.

Type_num	**Returns**
1	Name of the environment in which Microsoft Excel is running, as text, followed by the environment's version number.
2	The version number of Microsoft Excel, as text (for example, "5.0").
3	If fixed decimals are set, returns the number of decimals; otherwise, returns 0.
4	If in R1C1 mode, returns TRUE; if in A1 mode, returns FALSE.
5	If scroll bars are displayed, returns TRUE; otherwise, returns FALSE. See also GET.WINDOW(26) and GET.WINDOW(27).
6	If the status bar is displayed, returns TRUE; otherwise, returns FALSE.
7	If the formula bar is displayed, returns TRUE; otherwise, returns FALSE.
8	If remote DDE requests are enabled, returns TRUE; otherwise, returns FALSE.
9	Returns the alternate menu key as text; if no alternate menu key is set, returns the #N/A error value.
10	Number indicating special modes: 1 = Data Find 2 = Copy

	3 = Cut 4 = Data Entry 5 = Unused 6 = Copy and Data Entry 7 = Cut and Data Entry If no special mode is set, returns 0.
11	X position of the Microsoft Excel workspace window, measured in points from the left edge of the screen to the left edge of the window. In Microsoft Excel for the Macintosh, always returns 0.
12	Y position of the Microsoft Excel workspace window, measured in points from the top edge of the screen to the top edge of the window. In Microsoft Excel for the Macintosh, always returns 0.
13	Usable workspace width, in points.
14	Usable workspace height, in points.
15	Number indicating maximized or minimized status of Microsoft Excel: 1 = Neither 2 = Minimized 3 = Maximized Microsoft Excel for the Macintosh always returns 3.
16	Amount of memory free (in kilobytes).
17	Total memory available to Microsoft Excel (in kilobytes).
18	If a math coprocessor is present, returns TRUE; otherwise, returns FALSE.
19	If a mouse is present, returns TRUE; otherwise, returns FALSE. In Microsoft Excel for the Macintosh, always returns TRUE.
20	If a group is present in the workspace, returns a horizontal array of sheets in the group; otherwise returns the #N/A error value.

21 If the Standard toolbar is displayed, returns TRUE; otherwise, returns FALSE.

22 DDE-application-specific error code.

23 Full path of the default startup directory or folder.

24 Full path of the alternate startup directory or folder; returns the #N/A error value if no alternate path has been specified.

25 If Microsoft Excel is set for relative recording, returns TRUE; if set for absolute recording, returns FALSE.

26 Name of user.

27 Name of organization.

28 If Microsoft Excel menus are switched to by the transition menu or help key, returns 1; if Lotus 1-2-3 Help is switched to, returns 2.

29 If transition navigation keys are enabled, returns TRUE.

30 A nine-item horizontal array of global (default) print settings that can be set by the LINE.PRINT function:

Setup text

Left margin

Right margin

Top margin

Bottom margin

Page length

Logical value indicating whether to wait after printing each page (TRUE) or use continuous form feeding (FALSE)

Logical value indicating whether the printer has automatic line feeding (TRUE) or requires line feed characters (FALSE)

The number of the printer port

31 If a currently running macro is in single step mode, returns TRUE; otherwise, returns FALSE.

32 The current location of Microsoft Excel as a complete path.

33 A horizontal array of the names in the New list, in the order they appear.

34 A horizontal array of template files (with complete paths) in the New list, in the order they appear (returns the names of custom template files and the #N/A error value for built-in document types).

35 If a macro is paused, returns TRUE; FALSE otherwise.

36 If the Allow Cell Drag And Drop check box is selected in the Edit tab of the Options dialog box that appears when you click the Options command on the Tools menu, returns TRUE; otherwise, returns FALSE.

37 A 45-item horizontal array of the items related to country versions and settings. Use the following macro formula to return a specific item, where number is a number in the list below:

```
INDEX(GET.WORKSPACE(37), number)
```

These values apply to country codes:
1 = Number corresponding to the country version of Microsoft Excel.
2 = Number corresponding to the current country setting in the Microsoft Windows Control Panel or the country number as determined by your Apple system software

These values apply to number separators:
3 = Decimal separator
4 = Zero (or 1000) separator
5 = List separator

These values apply to R1C1-style references:
6 = Row character
7 = Column character
8 = Lowercase row character
9 = Lowercase column character
10 = Character used instead of the left bracket ([)
11 = Character used instead of the right bracket (])

These values apply to array characters:
12 = Character used instead of the left bracket ({)
13 = Character used instead of the right bracket (})
14 = Column separator

15 = Row separator
16 = Alternate array item separator to use if the current array separator is the same as the decimal separator

These values apply to format code symbols:
17 = Date separator
18 = Time separator
19 = Year symbol
20 = Month symbol
21 = Day symbol
22 = Hour symbol
23 = Minute symbol
24 = Second symbol
25 = Currency symbol
26 = "General" symbol

These values apply to format codes:
27 = Number of decimal digits to use in currency formats
28 = Number indicating the current format for negative currencies:
  0 = ($currency) or (currency$)
  1 = -$currency or -currency$
  2 = $-currency or currency-$
  3 = $currency- or currency$-
where currency is any number and the $ represents the current currency symbol.
29 = Number of decimal digits to use in noncurrency number formats
30 = Number of characters to use in month names
31 = Number of characters to use in weekday names
32 = Number indicating the date order:
  0 = Month-Day-Year
  1 = Day-Month-Year
  2 = Year-Month-Day

These values apply to logical format values:
33 = TRUE if using 24-hour time; FALSE if using 12-hour time.
34 = TRUE if not displaying functions in English; otherwise, returns FALSE.
35 = TRUE if using the metric system; FALSE if using the English measurement system.
36 = TRUE if a space is added before the currency symbol; otherwise, returns FALSE.
37 = TRUE if currency symbol precedes currency values; FALSE if it follows currency values.
38 = TRUE if using minus sign for negative numbers; FALSE if using parentheses.
39 = TRUE if trailing zeros are displayed for zero currency values; otherwise, returns FALSE.

	40 = TRUE if leading zeros are displayed for zero currency values; otherwise, returns FALSE. 41 = TRUE if leading zero is displayed in months (when months are displayed as numbers); otherwise, returns FALSE. 42 = TRUE if leading zero is shown in days (when days are displayed as numbers); otherwise, returns FALSE. 43 = TRUE if using four-digit years; FALSE if using two-digit years. 44 = TRUE if date order is month-day-year when displaying dates in long form; FALSE if date order is day-month-year. 45 = TRUE if leading zero is shown in the time; otherwise, returns FALSE.
38	The number 0, 1, or 2 indicating the type of error-checking as set by the ERROR function. For more information, see ERROR.
39	A reference in R1C1-text form to the currently defined error-handling macro (set by the ERROR function), or the #N/A error value if none is specified.
40	If screen updating is turned on (set by the ECHO function), returns TRUE; otherwise, returns FALSE.
41	A horizontal array of cell ranges, as R1C1-style text, that were previously selected with the Go To command from the Edit menu or the FORMULA.GOTO macro function. If the book has multiple sheets, or if the single sheet in the workbook is named differently than the workbook itself, returns names as [Book]Sheet.
42	If your computer is capable of playing sounds, returns TRUE; otherwise, returns FALSE.
43	If your computer is capable of recording sounds, returns TRUE; otherwise, returns FALSE.
44	A three-column array of all currently registered procedures in dynamic link libraries (DLLs). The first column contains the names of the DLLs that contain the procedures (in Microsoft Excel for Windows) or the names of the files that contain the code resources (in Microsoft Excel for the Macintosh). The second column contains the names of the procedures in the DLLs (in Microsoft Excel for Windows) or code resources (in Microsoft Excel for the Macintosh). The third column contains text strings specifying the data types of the return values, and the number and data types of the arguments. For more information about DLLs and code resources and data types, see Using the CALL and REGISTER functions in Microsoft Excel Help.

45	If Microsoft Windows for Pen Computing is running, returns TRUE; otherwise, returns FALSE.
46	If the Move Selection After Enter check box is selected in the Edit tab of the Options dialog box, returns TRUE; otherwise, returns FALSE.
47	Reserved.
48	Path to the library subdirectory for Microsoft Excel, as text.
49	MAPI session currently in use, returned as a string of hex digits encoding the mail session value.
50	If the Full Screen mode is on, returns TRUE; otherwise, FALSE.
51	If the formula bar is displayed in Full Screen mode, returns TRUE; otherwise, FALSE.
52	If the status bar is displayed in Full Screen mode, returns TRUE; otherwise, FALSE.
53	The name of the topmost custom dialog sheet currently running in a modal window, or #N/A if no dialog sheet is currently running.
54	If the Edit Directly In Cell check box is selected on the Edit tab in the Options dialog box, returns TRUE; otherwise, returns FALSE.
55	TRUE if the Alert Before Overwriting Cells check box in the Edit tab on Options dialog box is selected; otherwise, FALSE.
56	Standard font name in the General tab in the Options dialog box, as text.
57	Standard font size in the General tab in the Options dialog box, as a number
58	If the Recently Used File list check box in the General tab on the Options dialog box is selected, returns TRUE; otherwise, FALSE.
59	If the Display Old Menus check box in the General tab on the Options dialog box is selected, returns TRUE; otherwise, FALSE.
60	If the Tip Wizard is enabled, returns TRUE; otherwise, FALSE.
61	Number of custom list entries listed in the Custom Lists tab of the Options dialog box.
62	Returns information about available file converters.

63	Returns the type of mail system in use by Excel: 0 = No mail transport detected 1 = MAPI based transport 2 = PowerTalk based transport (Macintosh only)
64	If the Ask To Update Automatic Links check box in the Edit tab of the Options dialog box is selected, returns TRUE; otherwise, FALSE.
65	If the Cut, Copy, And Sort Objects With Cells check box in the Edit tab on the Options dialog box is selected, returns TRUE; otherwise, FALSE.
66	Default number of sheets in a new workbook, as a number, from the General tab on Options dialog box.
67	Default file directory location, as text, from the General tab in the Options dialog box.
68	If the Show ScreenTips On Toolbars check box in the Options tab in the Customize dialog box is selected, returns TRUE; otherwise, FALSE.
69	If the Large Icons check box in the Options tab in the Customize dialog box is selected, returns TRUE; otherwise, FALSE.
70	If the Prompt For Workbook Properties check box in the General tab on the Options dialog box is selected, returns TRUE; otherwise, FALSE.
71	TRUE if Microsoft Excel is open for in-place object editing (OLE). If FALSE, it is opened normally.
72	TRUE if the Color Toolbars check box is selected in the Toolbars dialog box. FALSE if the Color Toolbars check box is not selected. This argument is for compatibility with Microsoft Excel version 5.0.

Related Functions

GET.DOCUMENT   Returns information about a workbook

GET.WINDOW   Returns information about a window

GOAL.SEEK

Equivalent to clicking the Goal Seek command on the Tools menu. Calculates the values necessary to achieve a specific goal. If the goal is an amount returned by a formula, the GOAL.SEEK function calculates values that, when supplied to your formula, cause your formula to return the amount you want.

Syntax

GOAL.SEEK(**target_cell, target_value, variable_cell**)

GOAL.SEEK?(target_cell, target_value, variable_cell)

Target_cell   corresponds to the Set Cell box in the Goal Seek dialog box and is a reference to the cell containing the formula. If target_cell does not contain a formula, Microsoft Excel displays an error message.

Target_value   corresponds to the To Value box in the Goal Seek dialog box and is the value you want the formula in target_cell to return. This value is called a goal.

Variable_cell   corresponds to the By Changing Cell box in the Goal Seek dialog box and is the single cell that you want Microsoft Excel to change so that the formula in target_cell returns target_value. Target_cell must depend on variable_cell; if it does not, Microsoft Excel will not be able to find a solution.

Remarks

The max_num and max_change values set with the CALCULATION function can be used to change the solution process. Max_num sets the number of iterations; max_change determines the precision of the solution.

Tip   You can also use Microsoft Excel Solver to help solve your math equations for optimal values.

Related Functions

Related functions include the SOLVER functions, such as SOLVER.OPTIONS, SOLVER.SOLVE, and so on.

GOTO

Directs a macro to continue running at the upper-left cell of reference. Use GOTO to direct macro execution to another cell or a named range.

Syntax

GOTO(**reference**)

Reference   is a cell reference or a name that is defined as a reference. Reference can be an external reference to another macro sheet. If that macro sheet is not open, GOTO displays a message.

Tip   It's often preferable to use IF, ELSE, ELSE.IF, and END.IF instead of GOTO when you want to perform multiple actions based on a condition because the IF method makes your macros more structured.

Examples

If A1 contains the #N/A error value, then when the following formula is calculated, the macro branches to C3:

```
IF(ISERROR($A$1), GOTO($C$3))
```

You can also use macro names with GOTO statements. The following macro formula branches macro execution to a macro named Compile:

```
GOTO(Compile)
```

Because Compile is a named range, it should not be enclosed in quotation marks.

Related Function

FORMULA.GOTO   Selects a named area or reference on any open workbook

GRIDLINES

Allows you to turn chart gridlines on and off.

Arguments are logical values corresponding to the check boxes in the Gridlines dialog box. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box. If omitted, the setting is not changed. If a chart is not active, produces a error and halts the macro.

Syntax

GRIDLINES(x_major, x_minor, y_major, y_minor, z_major, z_minor, 2D_effect)

GRIDLINES?(x_major, x_minor, y_major, y_minor, z_major, z_minor, 2D_effect)

X_major   corresponds to the Category (X) Axis: Major Gridlines check box.

X_minor   corresponds to the Category (X) Axis: Minor Gridlines check box.

Y_major   corresponds to the Value (Y) Axis: Major Gridlines check box. On 3-D charts, y_major corresponds to the Series (Y) Axis: Major Gridlines check box.

Y_minor   corresponds to the Value (Y) Axis: Minor Gridlines check box. On 3-D charts, y_minor corresponds to the Series (Y) Axis: Minor Gridlines check box.

Z_major   corresponds to the Value (Z) Axis: Major Gridlines check box (3-D only).

Z_minor   corresponds to the Value (Z) Axis: Minor Gridlines check box (3-D only).

2D_effect   corresponds to the 2-D Walls and Gridlines check box (3-D only).

GROUP

Creates a single object from several selected objects and returns the object identifier of the group (for example, "Group 5"). Use GROUP to combine a number of objects so that you can move or resize them together.

If no object is selected, only one object is selected, or a group is already selected, GROUP returns the #VALUE! error value and interrupts the macro.

Syntax

GROUP( )

Related Function

UNGROUP   Separates a grouped object

HALT

Stops all macros from running. Use HALT instead of RETURN to prevent a macro from returning to the macro that called it.

Syntax

HALT(cancel_close)

Cancel_close is a logical value that specifies whether a macro sheet, when encountering the HALT function in an Auto_Close macro, is closed.

- If cancel_close is TRUE, Microsoft Excel halts the macro and prevents the workbook from being closed.
- If cancel_close is FALSE or omitted, Microsoft Excel halts the macro and allows the workbook to be closed.
- If cancel_close is specified in a macro that is not an Auto_Close macro, it is ignored and the HALT function simply stops the current macro.

Remarks

You can prevent an Auto_Close or Auto_Open macro from running by holding down the SHIFT key while opening or closing the workbook.

Examples

If A1 contains the #N/A error value, then when the following macro formula is calculated, the macro halts:

```
IF(ISERROR(A1), HALT(), GOTO(D4))
```

The following macro formula at the end of an Auto_Close macro ends the macro and prevents the workbook from being closed:

```
HALT(TRUE)
```

Related Functions

BREAK	Interrupts a FOR-NEXT, FOR.CELL-NEXT, or WHILE-NEXT loop
RETURN	Ends the currently running macro

HELP

Starts or switches to Help and displays the specified custom Help topic. Use HELP with custom Help files to create your own Help system, which can be used just like the built-in Microsoft Excel Help.

Syntax

HELP(help_ref)

Help_ref is a reference to a topic in a Help file, in the form "filename!topic_number".

- Help_ref must be given as text.

Remarks

- Microsoft Excel for Windows does not support the use of Help files in the text file format for custom Help.
- In Microsoft Excel for the Macintosh, custom Help files are plain text files or text files with line breaks.

Tips

- In Microsoft Excel for Windows, the following macro formula switches back to Microsoft Excel when Help is active:

```
APP.ACTIVATE()
```

- The following macro formula closes Help when Help is active:

```
SEND.KEYS("%{F4}")
```

Examples

In Microsoft Excel for Windows, the following macro formula displays the Help topic numbered 101 in the file CUSTHELP.DOC. The Help window remains open if the user switches to another window or application.

```
HELP("CUSTHELP.DOC!101")
```

If the custom Help file is not in the current directory, specify the full path along with the name of the file. For example:

```
HELP("C:\EXCEL\CUSTHELP.DOC!101")
```

In Microsoft Excel for the Macintosh, the following macro formula displays the Help topic numbered 101 in the file CUSTOM HELP:

```
HELP("CUSTOM HELP!101")
```

If the custom Help file is not in the current folder, specify the full path along with the name of the file. For example:

```
HELP("HARD DISK:EXCEL:HELP:CUSTOM HELP!101")
```

HIDE

Equivalent to clicking the Hide command on the Window menu. Hides the active window.

Syntax

HIDE( )

Tip Hiding windows can speed up your macros. You can switch to hidden windows with the ACTIVATE function. You can continue to use functions that refer to specific sheets, such as FORMULA and the GET functions, even when those sheets are hidden.

Related Function

UNHIDE Displays a hidden window

HIDE.DIALOG

Closes the dialog box that has the current focus.

Syntax

HIDE.DIALOG(cancel_logical)

Cancel_logical is a logical value that specifies whether to close the dialog box and validate any edit boxes. If FALSE, the dialog box is closed and edit boxes are validated, or checked to determine if they contain a valid data type. If TRUE, the dialog box is closed and the edit boxes are not validated.

Remarks

If the edit box does not contain a valid data type when the dialog box is closed, the dialog will remain open. For example, if the edit box is supposed to contain integer values, and a text value is entered, the dialog box will not close. This applies to only those dialog boxes that must be closed before any further user action can happen.

Examples

`HIDE.DIALOG(FALSE)` closes the dialog box and checks to see if the edit box contains a valid data type (validated)

Related Functions

EDITBOX.PROPERTIES Sets the properties of an edit box on a worksheet or dialog sheet

SHOW.DIALOG Runs a dialog on a dialog sheet

HIDE.OBJECT

Hides or displays the specified object.

Syntax

HIDE.OBJECT(object_id_text, hide)

Object_id_text is the name and number, or number alone, of the object, as text, as it appears in the reference area when the object is selected. The name of the object is also the text returned by the CREATE.OBJECT function, so object_id_text can be a reference to a cell containing CREATE.OBJECT. To give the name of more than one object, use the following format for object_id_text:

```
"oval 3, text 2, arc 5"
```

If object_id_text is omitted, the function operates on all selected objects. If no object is selected or if the object specified by object_id_text does not exist, HIDE.OBJECT returns the #VALUE! error value.

Hide is a logical value that specifies whether to hide or display the specified object. If hide is TRUE or omitted, Microsoft Excel hides the object; if FALSE, Microsoft Excel displays the object.

Remarks

Objects are not automatically selected after they are unhidden.

Examples

The following macro formula hides the selected object:

```
HIDE.OBJECT(, TRUE)
```

The following macro formula displays the object named Oval 3:

```
HIDE.OBJECT("Oval 3", FALSE)
```

The following macro formula displays the three specified objects:

```
HIDE.OBJECT("oval 3, text 2, arc 5", FALSE)
```

Related Functions

CREATE.OBJECT Creates an object

DISPLAY Controls how an object is displayed

HISTOGRAM

Calculates individual and cumulative percentages for a range of data and a corresponding range of data bins.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

HISTOGRAM(**inprng**, outrng, binrng, pareto, chartc, chart, labels)

HISTOGRAM?(inprng, outrng, binrng, pareto, chartc, chart, labels)

Inprng is the input range.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Binrng is an optional set of numbers that define the bin ranges. The values must be in ascending order. The values are interpreted as more than value A up to value B, more than value B up to value C, and so on. One additional bin is created for values for less than the minimum value specified in binrng.

Pareto is a logical value.

- If pareto is TRUE, data in the output table is presented in both ascending-bin order and descending-frequency order.
- If pareto is FALSE or omitted, data in the output table is presented in ascending-bin order only.

Chartc is a logical value. If chartc is TRUE, HISTOGRAM generates a cumulative percentages column in the output table. If both chartc and chart are TRUE, HISTOGRAM also includes a cumulative percentage line in the histogram chart. If omitted, chartc is FALSE.

Chart is a logical value. If chart is TRUE, HISTOGRAM generates a histogram chart in addition to the output table. If omitted, chart is FALSE.

Labels is a logical value.

- If labels is TRUE, then the first row or column of inprng contains labels.
- If labels is FALSE or omitted, all cells in inprng are considered data. Microsoft Excel generates appropriate data labels for the output table.

HLINE

Scrolls through the active window by a specific number of columns. Returns the #VALUE! error value if the active sheet is a chart.

Syntax

HLINE(**num_columns**)

Num_columns is the number of columns in the active worksheet or macro sheet you want to scroll through horizontally.

- If num_columns is positive, HLINE scrolls to the right.
- If num_columns is negative, HLINE scrolls to the left.
- Num_columns must be between -256 and 256, inclusive.

Example

The following function scrolls the active window by one-half window to the right:

```
HLINE(GET.WINDOW(15)/2)
```

Related Functions

HPAGE Horizontally scrolls through the active window one window at a time

HSCROLL Horizontally scrolls through a sheet by percentage or by column number

VLINE   Vertically scrolls through the active window by rows

VPAGE   Vertically scrolls through the active window one window at a time

VSCROLL   Vertically scrolls through a sheet by percentage or by row number

HPAGE

Horizontally scrolls through the active window one window at a time. Use HPAGE to change the displayed area of a worksheet or macro sheet.

Syntax

HPAGE(**num_windows**)

Num_windows   specifies the number of windows to scroll through the active window horizontally. A window is defined as the number of visible columns. If three columns are visible in the window, HPAGE scrolls through in increments of three columns.

- If num_windows is positive, HPAGE scrolls to the right.
- If num_windows is negative, HPAGE scrolls to the left.

Related Functions

HLINE   Horizontally scrolls through the active window by columns

HSCROLL   Horizontally scrolls through a worksheet by percentage or by column number

VLINE   Vertically scrolls through the active window by rows

VPAGE   Vertically scrolls through the active window one window at a time

VSCROLL   Vertically scrolls through a worksheet by percentage or by row number

HSCROLL

Horizontally scrolls through the active sheet by percentage or by column number.

Syntax

HSCROLL(**position**, col_logical)

Position   specifies the column you want to scroll to. Position can be an integer representing the column number or a fraction or percentage representing the horizontal position of the column in the sheet. If position is 0, HSCROLL scrolls through your sheet to its leftmost edge. If position is 1, HSCROLL scrolls through your sheet to its rightmost edge. For charts that do not size with the window, use a fraction or percentage.

Col_logical   is a logical value specifying how the function scrolls.

- If col_logical is TRUE, HSCROLL scrolls through the sheet to column position.
- If col_logical is FALSE or omitted, then HSCROLL scrolls through the sheet to the horizontal position represented by the fraction position.

Remarks

- To scroll to a specific column n, either use HSCROLL(n, TRUE) or use HSCROLL(n/256). To scroll to column 38, for example, use HSCROLL(38, TRUE) or HSCROLL(38/256).
- If you are recording a macro and move the scroll box several times in a row, the recorder only records the final location of the scroll box, omitting any intermediate steps. Remember that scrolling does not change the active cell or the selection.

Related Functions

HLINE	Horizontally scrolls through the active window by columns
HPAGE	Horizontally scrolls through the active window one window at a time
VLINE	Vertically scrolls through the active window by rows
VPAGE	Vertically scrolls through the active window one window at a time
VSCROLL	Vertically scrolls through a sheet by percentage or row number

IF

Used with ELSE, ELSE.IF, and END.IF to control which formulas in a macro are executed. There are two syntax forms of the IF function. The following syntax can be used on macro sheets only; use it when you want your macro to branch to a particular set of functions based on the outcome of a logical test. The worksheet form of this function can be used on worksheets and macro sheets.

Syntax

IF(**logical_test**)

Logical_test is a logical value that IF uses to determine which functions to carry out next—that is, where to branch.

- If logical_test is TRUE, Microsoft Excel carries out the functions between the IF function and the next ELSE, ELSE.IF, or END.IF function. Instructions between ELSE.IF or ELSE and END.IF are not carried out.
- If logical_test is FALSE, Microsoft Excel immediately branches to the next ELSE.IF, ELSE, or END.IF function.
- If logical_test produces an error, the macro halts.

TIPS

- Use IF with ELSE, ELSE.IF, and END.IF when you want to perform multiple actions based on a condition. This method is preferable to using GOTO because it makes your macros more structured.
- If your macro ends with an error at a cell containing this form of the IF function, make sure there is a corresponding END.IF function.

Example

The following macro runs the macro CompleteEntry if the user clicks OK:

```
IF(ALERT("Are you done with this entry?", 1), CompleteEntry(), )
```

Tip You can indent formulas in a macro. To indent a formula, type as many spaces as you want between the equal sign and the first letter of the formula.

Related Functions

ELSE Specifies an action to take if an IF function returns FALSE

ELSE.IF Specifies an action to take if an IF or another ELSE.IF function returns FALSE

END.IF Ends a group of macro functions started with an IF statement

ERROR Specifies what action to take if an error occurs while a macro is running

INITIATE

Opens a dynamic data exchange (DDE) channel to an application and returns the number of the open channel. Once you have opened a channel to another application with INITIATE, you can use EXECUTE and SEND.KEYS to control the other application from a Microsoft Excel macro. (SEND.KEYS is available only with Microsoft Excel for Windows.) If INITIATE is successful, it returns the number of the open channel. All the subsequent DDE macro functions use this number to specify the channel. If INITIATE is unsuccessful, FALSE is returned.

Important Microsoft Excel for the Macintosh requires system software version 7.0 or later for this function.

Syntax

INITIATE(**app_text, topic_text**)

App_text is the DDE name of the application with which you want to begin a DDE session, in text form. The form of app_text depends on the application you are accessing. The DDE name of Microsoft Excel, for example, is "Excel".

Topic_text describes something, such as a document or a record in a database, in the application that you are accessing; the form of topic_text depends on the application you are accessing. Microsoft Excel accepts the names of the current documents as topic_text, as well as the name "System".

Remarks

- You can specify an instance of an application by appending the application's task ID number to the app_text argument. If you start an application by using the EXEC function, EXEC returns the task ID number for that instance of the application.

- If more than one instance of an application is running and you do not specify which instance you would like to open a channel to, INITIATE displays a dialog box from which you can choose the instance you want. You can prevent this dialog box from appearing by disabling or redirecting errors with the ERROR function.

Example

The following macro formula opens a channel to the document named MEMO in the application named WORD:

```
INITIATE("WORD", "MEMO")
```

Related Functions

POKE	Sends data to another application
REQUEST	Returns data from another application
TERMINATE	Closes a channel to another application
EXECUTE	Carries out a command in another application
EXEC	Starts a separate program

INPUT

Displays a dialog box for user input. Returns the information entered in the dialog box. Use INPUT to display a simple dialog box for the user to enter information to be used in a macro.

The dialog box has an OK and a Cancel button. If you click the OK button, INPUT returns the default value specified or the value typed in the edit box. If you click the Cancel button, INPUT returns FALSE.

Syntax

INPUT(**message_text**, type_num, title_text, default, x_pos, y_pos, help_ref)

Message_text   is the text to be displayed in the dialog box. Message_text must be enclosed in quotation marks.

Type_num   is a number specifying the type of data to be entered.

Type_num	Data type
0	Formula
1	Number
2	Text
4	Logical
8	Reference
16	Error

64	Array

You can also use the sum of the allowable data types for type_num. For example, for an input box that can accept formulas, text, or numbers, set type_num equal to 3 (the sum of 0, 1, and 2, which are the type specifiers for formula, number, and text). If type_num is omitted, it is assumed to be 2.

- If type_num is 0, INPUT returns the formula in the form of text, for example, "=2*PI()/360".
- To enter a formula, include an equal sign at the beginning of the formula; otherwise the formula is returned as text.
- If the formula contains references, they are returned as R1C1-style references, for example, "=RC[-1]*(1+R1C1)".
- If type_num is 8, INPUT returns an absolute reference to the specified cells.
- If you enter a single-cell reference in the dialog box, the value in that cell is returned by the INPUT function.
- If the information entered in the dialog box is not of the correct data type, Microsoft Excel attempts to convert it to the specified type. If the information can't be converted, Microsoft Excel displays an error message.

Title_text is text specifying a title to be displayed in the title bar of the dialog box. If title_text is omitted, it is assumed to be "Input".

Default specifies a value to be shown in the edit box when the dialog box is initially displayed. If default is omitted, the edit box is left empty.

X_pos, y_pos specify the horizontal and vertical position, in points, of the dialog box. A point is 1/72nd of an inch. If either or both arguments are omitted, the dialog box is centered in the corresponding direction.

Help_ref is a reference to a custom online Help topic in a text file, in the form "filename!topic_number".

- If help_ref is present, a Help button appears in the lower-right corner of the dialog box. Clicking the Help button starts Help and displays the specified topic.
- If help_ref is omitted, no Help button appears.
- Help_ref must be given as text.

For more information about custom Help topics, see HELP.

Remarks

Relative references entered in formulas in the INPUT dialog box are relative to the active cell at the time the INPUT function is calculated. If you are using the reference entered into the dialog box in a cell other than the active cell, it may not refer to the cells you intend it to. For example, if the active cell is A3 and you enter the formula "=A1+A2" in an INPUT dialog box, intending to add the values in those cells, and then use the FORMULA function to enter the formula in cell B3, the formula in cell B3 will read "=B1+B2" because you gave a relative reference. You can use FORMULA.CONVERT to solve this problem.

Examples

In Microsoft Excel for Windows, the following macro formula displays the following dialog box:

```
INPUT("Enter the inflation rate:", 1, "Inflation Rate", , , ,
"CUSTHELP.DOC!101")
```

If you then enter 12%, INPUT returns the value 0.12.

In Microsoft Excel for the Macintosh, the following macro formula displays the following dialog box:

```
INPUT("Enter the inflation rate:", 1, "Inflation Rate", , , , "CUSTOM
HELP!101")
```

If you then enter 12%, INPUT returns the value 0.12.

If the active cell is C2 and you enter the formula =B2*(1+$A$1) in response to the following macro formula:

```
INPUT("Enter your monthly increase formula:", 0)
```

INPUT returns "=RC[-1]*(1+R1C1)"

If you select the range $A$2:$A$8 in the INPUT dialog box:

```
REFTEXT|USA|002|001|001|common|UREFTEXT(INPUT("Please make your
selection.", 8)) returns R2C1:R8C1
```

Related Functions

ALERT   Displays a dialog box and a message

DIALOG.BOX   Displays a custom dialog box

FORMULA.CONVERT   Changes the style and type of references in a formula

HELP   Displays a custom Help topic

INSERT

Inserts a blank cell or range of cells or pastes cells from the Clipboard into a sheet. Shifts the selected cells to accommodate the new ones. The size and shape of the inserted range are the same as those of the current selection.

Syntax

INSERT(shift_num)

INSERT?(shift_num)

Shift_num is a number from 1 to 4 specifying which way to shift the cells. If an entire row or column is selected, shift_num is ignored. If shift_num is omitted, Microsoft Excel shifts cells in the logical direction based on the selection.

Shift_num	Direction
1	Shift cells right
2	Shift cells down
3	Shift entire row
4	Shift entire column

Remarks

If you have just cut or copied information to the Clipboard, INSERT performs both an insert and a paste operation. First, Microsoft Excel inserts new blank cells into the sheet; then, Microsoft Excel pastes information from the Clipboard into the newly inserted cells. If you have used the INSERT function in macros written for Microsoft Excel version 2.2 or earlier, make sure you consider this feature when you use your old macros with later versions of Microsoft Excel.

Related Functions

COPY Copies and pastes data or objects

CUT Cuts or moves data or objects

EDIT.DELETE Removes cells from a sheet

PASTE Pastes cut or copied data

INSERT.OBJECT

Equivalent to choosing the Object command from the Insert menu, and then selecting an object type and choosing the OK button. Creates an embedded object whose source data is supplied by another application. Also starts an application of the appropriate class for the specified object type.

Syntax

INSERT.OBJECT(**object_class**, file_name, link_logical, display_icon_logical, icon_file, icon_number, icon_label)

INSERT.OBJECT?(object_class, file_name, link_logical, display_icon_logical, icon_file, icon_number, icon_label)

Object_class is a text string containing the classname for the object you want to create.

- Object_class is the classname corresponding to the Object Type selection in the Insert Object dialog box.

- For more information about object classnames, consult the documentation for your source application to see how it supports object linking and embedding (OLE).

File_name is a text string specifying the file from which to create an OLE object.

Link_logical is a logical value indicating whether the new object based on file_name should be linked to file_name. If it is not linked, the object is created as a copy or the file. Link_logical is ignored if file_name is not specified. If link_logical is FALSE or omitted, then no link is established.

Display_icon_logical is a logical value corresponding to the Display as Icon checkbox. If it is FALSE or omitted, then the regular picture for the object is displayed. If it is TRUE, then the icon icon_number found in icon_file is displayed with the label icon_label. If display_icon_logical is not TRUE, then icon_file, icon_number, and icon_label are ignored.

Icon_file is the name of the file where the icon to display is located.

Icon_number is the number of the icon within icon_file that should be used.

Icon_label is a text string indicating a label to display beneath the icon. If the parameter is an empty string ("") or is omitted, no label is displayed.

Remarks

- If INSERT.OBJECT starts another application, your macro pauses. Your macro resumes when you return to Microsoft Excel.

- Although you will not normally use Microsoft Excel class names in a Microsoft Excel macro, you may need them in macros written for other applications. Microsoft Excel uses classnames "Excel.Sheet.5" and "Excel.Chart.5".

Related Function

EDIT.OBJECT Edits an object

INSERT.PICTURE

Equivalent to clicking the Picture command on the Insert menu. This function is available for Microsoft Excel for Windows only .

Syntax

INSERT.PICTURE(file_name, filter_number)

INSERT.PICTURE?(file_name, filter_number)

File_name is the name, as text, of the file containing the picture that you want to insert into your workbook.

Filter_number is a number specifying which converter Microsoft Excel will use to open the file.

Convert_type	Converter and filename extension
1	Windows Bitmaps (bmp)
2	Windows Metafile (wmf)
3	DrawPerfect (wpg)
4	Micrografix Designer/Draw (drw)
5	AutoCAD Format 2-D (dxf)
6	HP Graphics Language (hgl)
7	Computer Graphics Metafile (cgm)
8	Encapsulated Postscript (eps)
9	Tagged Image Format (tif)
10	PC PaintBrush (pcx)
11	Lotus 1-2-3 Graphic (pic)
12	AutoCAD Plot Files (plt)
13	Macintosh PICT (pct)

INSERT.TITLE

Attaches text to various parts of a chart.

Syntax

2-D charts

INSERT.TITLE(chart, y_primary, x_primary, y_secondary, x_secondary)

3-D charts

INSERT.TITLE(chart, z_primary, x_primary, y_primary)

Chart is a logical value specifying whether to attach a title to the chart.

Y_primary is a logical value specifying whether to attach a title to the value (y) axis of a 2-D chart or the series (y) axis of a 3-D chart.

X_primary is a logical value specifying whether to attach a title to the category (x) axis of the chart.

Z_primary is a logical value specifying whether to attach a title to the value (z) axis of a 3-D chart.

Y_secondary is a logical value specifying whether to attach a title to the second value (y) axis of a chart containing more than one chart type.

X_secondary is a logical value specifying whether to attach a title to the second category (x) axis of a chart containing more than one chart type.

Remarks

To change the text in a selected title, use the FORMULA function.

Related Function

FORMULA Enters formulas in a chart

JUSTIFY

Equivalent to clicking the Justify command on the Fill submenu of the Edit menu. Rearranges the text in a range so that it fills the range evenly.

Syntax

JUSTIFY()

Related Function

ALIGNMENT Aligns the contents of the selected cells

LABEL.PROPERTIES

Sets the accelerator property of the label and group box controls on a worksheet or dialog sheet.

Syntax

LABEL.PROPERTIES(accel_text, accel_text2, 3d_shading)

LABEL.PROPERTIES?(accel_text, accel_text2, 3d_shading)

Accel_text is a text string containing the character to use as the label's accelerator key on a dialog sheet. The character is matched against the text of the control, and the first matching character is underlined. When the user presses ALT+accel_text in Microsoft Excel for Windows or COMMAND+accel_text in Microsoft Excel for the Macintosh, the control is clicked. This argument is ignored for controls on worksheets.

Accel_text2 is a text string containing the second accelerator key on a dialog sheet. This argument is for only Far East versions of Microsoft Excel.

3d_shading is a logical value that specifies whether the list box appears as 3-D. If TRUE, the list box will appear as 3-D. If FALSE or omitted, the list box will not be 3-D. This argument is available for only worksheets.

Related Functions

CHECKBOX.PROPERTIES Sets various properties of check box and option box controls

SCROLLBAR.PROPERTIES Sets the properties of the scroll bar and spinner controls

PUSHBUTTON.PROPERTIES Sets the properties of the push button control

LAST.ERROR

Returns the reference to the cell where the last macro sheet error occurred. If no error has occurred, LAST.ERROR returns the #N/A error value. Use LAST.ERROR in conjunction with the REFTEXT function to quickly locate errors.

Syntax

LAST.ERROR( )

Related Function

ERROR Specifies what action to take if an error is encountered while a macro is running

LEGEND

Adds a legend to or removes a legend from a chart. This is also equivalent to clicking the Legend button on the Chart toolbar when a chart is active.

Syntax

LEGEND(logical)

Logical is a logical value specifying which command LEGEND is equivalent to.

- If logical is TRUE or omitted, LEGEND is equivalent to the Legend command on the Insert menu.
- If logical is FALSE, LEGEND is equivalent to the Delete command on the Edit menu.
- If logical is FALSE and the active chart has no legend, LEGEND takes no action.

Related Function

FORMAT.LEGEND Determines the position and orientation of the legend on a chart

LINE.PRINT

Prints the active worksheet using methods compatible with those of Lotus 1-2-3. LINE.PRINT does not use the Microsoft Windows printer drivers. Unless you have a specific need for the LINE.PRINT function, use the PRINT function instead.

Note This function is only available in Microsoft Excel for Windows.

Syntax 1

Go, Line, Page, Align, and Clear

LINE.PRINT(**command**, file, append)

Syntax 2

Worksheet settings

LINE.PRINT(**command**, setup_text, leftmarg, rightmarg, topmarg, botmarg, pglen, formatted)

Syntax 3

Global settings

LINE.PRINT(**command**, setup_text, leftmarg, rightmarg, topmarg, botmarg, pglen, wait, autolf, port, update)

Command is a number corresponding to the command you want LINE.PRINT to carry out. For syntax 2, command must be 5. For syntax 3, command must be 6.

Command	**Command that is carried out**
1	Go
2	Line
3	Page
4	Align
5	Worksheet settings
6	Global settings (saved in EXCEL5.INI)
7	Clear (change to current global settings)

File is the name of a file to which you want to print. If omitted, Microsoft Excel prints to the printer port determined by the current global settings.

Append is a logical value specifying whether to append text to file. If TRUE, the file you are printing is appended to file; if FALSE or omitted, the file you are printing overwrites the contents of file.

Setup_text is text that includes a printer initialization sequence or other control codes to prepare your printer for printing. If omitted, no setup text is used.

Leftmarg is the size of the left margin measured in characters from the left side of the page. If omitted, it is assumed to be 4.

Rightmarg is the size of the right margin measured in characters from the left side of the page. If omitted, it is assumed to be 76.

Topmarg is the size of the top margin measured in lines from the top of the page. If omitted, it is assumed to be 2.

Botmarg is the size of the bottom margin measured in lines from the bottom of the page. If omitted, it is assumed to be 2.

Pglen is the number of lines on one page. If omitted, it is assumed to be 66 (11 inches with 6 lines per inch). If you're using an HP LaserJet or compatible printer, set pglen to 60 (the printer reserves six lines).

Formatted is a logical value specifying whether to format the output. If TRUE or omitted, the output is formatted; if FALSE, it is not formatted.

Wait is a logical value specifying whether to wait after printing a page. If TRUE, Microsoft Excel waits; if FALSE or omitted, Microsoft Excel continues printing.

Autolf is a logical value specifying whether your printer has automatic line feeding. If TRUE, Microsoft Excel prints lines normally; if FALSE or omitted, Microsoft Excel sends an additional line feed character after printing each line.

Port is a number from 1 to 8 specifying which port to use when printing.

Port	**Port used when printing**
1 or omitted	LPT1
2	COM1
3	LPT2
4	COM2
5	LPT1
6	LPT2
7	LPT3

8	LPT4

Update is a logical value specifying whether to update and save global settings. If TRUE, the settings are saved in the EXCEL5.INI file; if FALSE or omitted, the global settings are not saved.

Remarks

The default values for print settings on your worksheet are determined by the current global settings.

Example

The following macro formula prints the currently defined print area to the currently defined printer port:

```
LINE.PRINT(1)
```

Related Function

PRINT Prints the active sheet

LINK.COMBO

Links an edit box and a list box control into a linked combination box group. The resulting linked controls track each other's selection and contents. Linked edit and list box combinations are similar to an editable drop-down list box, except that the list box is permanently visible and dropped down.

Syntax

LINK.COMBO(**link_logical**)

Link_logical is a logical value that specifies whether the controls are linked or unlinked. If TRUE, the controls will become linked. If FALSE, the controls will be unlinked.

Remarks

To use this function, first select the list box and edit box to be linked or unlinked. You can do this with `SELECT("list box 1,Edit box 2")`.

Examples

`LINK.COMBO(FALSE)` will unlink a list box and an edit box.

Related Functions

ADD.LIST.ITEM Adds an item in a list box or drop-down control on a worksheet or dialog sheet control

SELECT.LIST.ITEM Selects an item in a list box or in a group box

LINK.FORMAT

Links the number format of the selected data label to the worksheet cell or range containing the data label text.

Syntax

LINK.FORMAT( )

LINKS

Returns, as a horizontal array of text values, the names of all workbooks referred to by external references in the workbook specified. Use LINKS with OPEN.LINKS to open supporting workbooks.

Syntax

LINKS(document_text, type_num)

Document_text is the name of a workbook, including its path. If document_text is omitted, LINKS operates on the active workbook. If the workbook specified by document_text is not open, LINKS returns the #N/A error value.

Type_num is a number from 1 to 6 specifying the type of linked workbooks to return.

Type_num	Returns
1 or omitted	Microsoft Excel link
2	DDE/OLE link (Microsoft Excel for Windows)
3	Reserved
4	Not applicable
5	Publisher (Microsoft Excel for the Macintosh)
6	Subscriber (Microsoft Excel for the Macintosh)

Remarks

- If the active workbook contains no external references, LINKS returns the #N/A error value.
- With the INDEX function, you can select individual workbook names from the array for use in other functions that take workbook names as arguments.

- The names of the workbook are always returned in alphabetic order. If supporting workbooks are open, LINKS returns the names of the workbooks; if supporting workbooks are closed, LINKS includes the full path of each workbook.

- If type_num is 5 or 6, LINKS returns a two-row array in which the first row contains the edition name and the second row contains the reference.

Examples

If a chart named Chart1 is open and contains links to workbook Data1 and Data2, and the LINKS function shown below is entered as an array into a two-cell horizontal range:

`LINKS("Chart1")` equals "Data1" in the first cell of the range and "Data2" in the second cell.

In Microsoft Excel for Windows, if the chart named VARIANCE.XLS is open and contains data series that refer to workbook named BUDGET.XLS and ACTUAL.XLS, then:

`OPEN.LINKS(LINKS("VARIANCE.XLS"))` opens BUDGET.XLS and ACTUAL.XLS.

In Microsoft Excel for the Macintosh, if the workbook named SALES 1991 is open and contains references to the workbook WEST SALES, SOUTH SALES, and EAST SALES, then:

`OPEN.LINKS(LINKS("SALES 1991"))` opens WEST SALES, SOUTH SALES, and EAST SALES.

Related Functions

CHANGE.LINK	Changes supporting workbook links
GET.LINK.INFO	Returns information about a link
OPEN.LINKS	Opens specified supporting workbook
UPDATE.LINK	Updates a link to another workbook

LISTBOX.PROPERTIES

Sets the properties of a list box and drop-down controls on a worksheet or dialog sheet.

Syntax

LISTBOX.PROPERTIES(range, link, drop_size, multi_select, 3d_shading)

LISTBOX.PROPERTIES?(range, link, drop_size, multi_select, 3d_shading)

Range is the cell range that the initial contents of the list box are taken from. If blank (empty text), the list box is initially unfilled.

Link is the cell on the sheet to which the list box is linked, and indicates the numeric position of the currently selected item in the list box. Whenever an item in the list box is selected, its numeric position is entered into the linked cell on the sheet.

Drop_size is the number of lines shown when a drop-down control is dropped. This value is ignored when applied to a non-drop-down list box.

Multi_select is a number that specifies the selection mode of the list box. Zero is single selection. 1 is simple multi-select. 2 is extended multi-select.

3d_shading is a logical value that specifies whether the list box appears as 3-D. If TRUE, the list box will appear as 3-D. If FALSE or omitted, the list box will not be 3-D. This argument is available for only worksheets.

Related Functions

ADD.LIST.ITEM   Adds an item in a list box or drop-down control on a worksheet or dialog sheet control

SELECT.LIST.ITEM   Selects an item in a list box or in a group box

CHECKBOX.PROPERTIES   Sets various properties of check box and option box controls

SCROLLBAR.PROPERTIES   Sets the properties of the scroll bar and spinner controls

PUSHBUTTON.PROPERTIES   Sets the properties of the push button control

LIST.NAMES

Equivalent to clicking the Paste command on the Name submenu of the Insert menu and selecting the Paste List option button. Lists all names (except hidden names) defined in your workbook. LIST.NAMES also lists the cells to which the names refer; whether a macro corresponding to a particular name is a command macro or a custom function; the shortcut key for each command macro; and the category of the custom functions.

Syntax

LIST.NAMES( )

Remarks

- If the current selection is a single cell or five or more columns wide, LIST.NAMES pastes all five types of information about worksheet names into five columns. The first column contains cell names. The second column contains the corresponding cell references. The third column contains the number 1 if the name refers to a custom function, the number 2 if it refers to a command macro, or 0 if it refers to anything else. The fourth column lists the shortcut keys for command macros. The fifth column contains a category name for custom functions or the number of the built-in category.

- If the selection includes fewer than five columns, LIST.NAMES omits the information that would have been pasted into the other columns.

- When you use LIST.NAMES, Microsoft Excel completely replaces the contents of the cells it pastes into.

Related Functions

GET.DEF   Returns a name matching a definition

GET.NAME   Returns the definition of a name

NAMES   Returns the names defined in a workbook

MACRO.OPTIONS

Equivalent to clicking the Options button in the Macro dialog box, which appears when you click the Macros command (Tools menu, Macro submenu).

Syntax

MACRO.OPTIONS(**macro_name**, description, menu_on, menu_text, shortcut_on, shortcut_key, function_category, status_bar_text, help_id, help_file)

Macro_name is the name of the macro that you want to set options for, including the name of the workbook and sheet containing the macro.

Description is the description of the macro displayed in the Macro dialog box.

Menu_on is a logical value indicating whether a menu item is automatically added for this macro. If TRUE, menu_text must be specified. If FALSE or omitted, no menu item is added. If the macro already has a menu item, setting this argument to FALSE removes the menu item.

Menu_text is the text of the menu item to be added for the macro. Ignored unless menu_on is TRUE.

Shortcut_on is a logical value indicating whether a shortcut key is assigned to the macro. If TRUE, shortcut_key must be specified. If FALSE or omitted, no shortcut key is assigned. If the macro already has a shortcut key, setting this argument to FALSE removes the shortcut key.

Shortcut_key is the letter of the shortcut key for the macro. Ignored if shortcut_key is FALSE.

Function_category is the number of the category in the Paste Function dialog box that the macro is assigned to. Categories are numbered starting at 1 for the category at the top of the list in the Paste Function dialog box.

Status_bar_text the text displayed in the status bar when a menu item or toolbar button assigned to this macro is clicked on. Be sure to enclose the text in quotes.

Help_id is the numerical ID for the help topic associated with this macro and any related menu items or toolbar buttons.

Help_file is the pathname of the help file for the macro.

MAIL.ADD.MAILER

Equivalent to clicking the Add Mailer command on the Mail submenu of the File menu. Adds a new PowerTalk mailer to the active workbook. Use this command to add addressing or subject information to a workbook that you want to send to another user.

Note This function is available on Macintosh computers with Microsoft Excel and Apple PowerTalk only.

Syntax

MAIL.ADD.MAILER( )

Remarks

If there is already a mailer, this command fails and returns the #VALUE! error value.

Related Function

MAIL.DELETE.MAILER   Deletes an existing mailer from the active workbook

MAIL.DELETE.MAILER

Equivalent to clicking the Delete Mailer command on the Mail submenu of the File menu. Deletes an existing mailer from the active workbook.

Note   This function is available on Macintosh computers with Microsoft Excel and Apple PowerTalk only.

Syntax

MAIL.DELETE.MAILER( )

Remarks

If there is no mailer, returns the #VALUE! error value.

Related Function

MAIL.ADD.MAILER   Adds a new PowerTalk mailer to the active workbook

MAIL.EDIT.MAILER

Equivalent to clicking the Mailer button when mailer is attached to the current workbook. Allows you to edit a PowerTalk mailer attached to the active workbook

Note   This function is available on Macintosh computers with Microsoft Excel and Apple PowerTalk only.

Syntax

MAIL.EDIT.MAILER(to_recipients, cc_recipients, bcc_recipients, subject, enclosures, which_address)

MAIL.EDIT.MAILER?(to_recipients, cc_recipients, bcc_recipients, subject, enclosures, which_address)

To_recipients   is the name of the person to whom you want to send the mail. The name should be given as text. To specify more than one name, give the list of names as an array.

Cc_recipients   is the name of those recipients to be carbon copied. A single name should be given as text. To specify more than one name, give the list of names as an array.

Bcc_recipients   is the name of the recipients to be added as blind carbon copies.

Subject   is a text string containing the subject text for the mail messages.

Enclosures   is an array of strings specifying enclosures as file names.

Which_address   indicates which type of address to use, as a text string, specifying the address type for all recipients. For example, "Fax".

Remarks

If there is no mailer, returns the #VALUE! error value.

Related Functions

MAIL.DELETE.MAILER   Adds a new PowerTalk mailer to the active workbook

MAIL.ADD.MAILER   Adds a new PowerTalk mailer to the active workbook

MAIL.FORWARD

Equivalent to clicking the Forward command on the Mail submenu of the File menu. Creates a new mailer to replace the previous version and brings up the mailer dialog.

Note   This function is available on Macintosh computers with Microsoft Excel and Apple PowerTalk only.

Syntax

MAIL.FORWARD( )

Remarks

- Returns the #VALUE! error value or #N/A if the current workbook has no mailer.
- This function is available only when the current workbook is open and has been received by PowerTalk with a piece of mail to forward.

Related Functions

MAIL.EDIT.MAILER   Allows you to edit a PowerTalk mailer attached to the active workbook

MAIL.DELETE.MAILER   Deletes a new PowerTalk mailer to the active workbook

MAIL.ADD.MAILER   Adds a new PowerTalk mailer to the active workbook

MAIL.LOGOFF

Ends the current mail session.

Important   To use MAIL.LOGOFF in Microsoft Excel for Windows, you must be using a mail client that supports the Messaging Applications Programming Interface (MAPI) or Vendor-Independent Messaging (VIM). The function is available for only Microsoft Excel for Windows.

Syntax

MAIL.LOGOFF( )

Remarks

Returns TRUE if the session was ended, or #VALUE! if there was no session.

MAIL.LOGON

Starts a mail session.

Important   To use MAIL.LOGON in Microsoft Excel for Windows, you must be using a mail client that supports the Messaging Applications Programming Interface (MAPI) or Vendor-Independent Messaging (VIM). The function is available for only Microsoft Excel for Windows.

Syntax

MAIL.LOGON(name_text, password_text, download_logical)

MAIL.LOGON?(name_text, password_text, download_logical)

Name_text is the username of the mail account or Microsoft Exchange profile name. If omitted, prompts for username.

Password_text is the password for the account. If omitted, prompts for password. Ignored when the dialog box form is used. This argument is ignored in Microsoft Exchange.

Download_logical specifies whether to download new mail. Use TRUE to download new mail; use FALSE or leave blank to skip downloading new mail.

Remarks

Returns FALSE if you cancel the dialog box or #VALUE! if you can't log on.

If you omit both name_text and password_text, Microsoft Excel tries to log on using an existing mail session.

Related Function

MAIL.LOGOFF Ends the current mail session

MAIL.NEXT.LETTER

Equivalent to clicking the Next Letter command on the Mail submenu of the File menu. Opens the oldest unread Microsoft Excel workbook from the In Tray as a new window.

Note This function is available on Macintosh computers with Microsoft Excel and Apple PowerTalk only.

Syntax

MAIL.NEXT.LETTER( )

Remarks

Returns #VALUE! on error, and #N/A if there are no more letters in the In Tray to open.

Related Functions

MAIL.EDIT.MAILER Allows you to edit a PowerTalk mailer attached to the active workbook

MAIL.DELETE.MAILER Adds a new PowerTalk mailer to the active workbook

MAIL.ADD.MAILER Adds a new PowerTalk mailer to the active workbook

MAIL.REPLY

Equivalent to clicking the Reply command on the Mail submenu of the File menu. Replies to the sender of the current letter.

Note This function is available on Macintosh computers with Microsoft Excel and Apple PowerTalk only.

Syntax

MAIL.REPLY( )

Remarks

- Returns the #VALUE! error value or #N/A if the current workbook has no mailer.
- The letter must currently be open.

Related Functions

MAIL.EDIT.MAILER Allows you to edit a PowerTalk mailer attached to the active workbook

MAIL.DELETE.MAILER Deletes a new PowerTalk mailer to the active workbook

MAIL.ADD.MAILER Adds a new PowerTalk mailer to the active workbook

MAIL.REPLY.ALL

Equivalent to clicking the Reply All command on the Mail submenu of the File menu. Replies to the sender and all recipients of the current letter.

Note This function is available on Macintosh computers with Microsoft Excel and Apple PowerTalk only.

Syntax

MAIL.REPLY.ALL( )

Remarks

Returns the #VALUE! error value or #N/A if the current workbook has no mailer.

Related Functions

MAIL.EDIT.MAILER Allows you to edit a PowerTalk mailer attached to the active workbook

MAIL.DELETE.MAILER Deletes a new PowerTalk mailer to the active workbook

MAIL.ADD.MAILER Adds a new PowerTalk mailer to the active workbook

MAIL.SEND.MAILER

Equivalent to clicking the Send Mailer command on the Mail submenu of the File menu. Sends a PowerTalk mailer.

Note This function is available on Macintosh computers with Microsoft Excel and Apple PowerTalk only.

Syntax

MAIL.SEND.MAILER( )

MAIN.CHART

Equivalent to clicking the Main Chart command on the Format menu when a chart sheet is active in Microsoft Excel version 2.2 or earlier. This function is included only for macro compatibility.

Syntax

MAIN.CHART(**type_num**, stack, 100, vary, overlap, drop, hilo, overlap%, cluster, angle)

MAIN.CHART?(type_num, stack, 100, vary, overlap, drop, hilo, overlap%, cluster, angle)

MAIN.CHART.TYPE

Equivalent to clicking the Main Chart Type command on the Chart menu in Microsoft Excel for the Macintosh version 1.5 or earlier. This function is included only for macro compatibility.

Syntax

MAIN.CHART.TYPE(**type_num**)

MCORREL

Returns a correlation matrix that measures the correlation between two or more data sets that are scaled to be independent of the unit of measurement.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

MCORREL(**inprng**, outrng, grouped, labels)

MCORREL?(inprng, outrng, grouped, labels)

Inprng is the input range.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Grouped is a text character that indicates whether the data in the input range is organized by row or column.

- If grouped is "C" or omitted, then the data is organized by column.
- If grouped is "R", then the data is organized by row.

Labels is a logical value that describes where the labels are located in the input range, as shown in the following table:

Labels	**Grouped**	**Labels are in**
TRUE	"C"	First row of the input range.
TRUE	"R"	First column of the input range.

FALSE or omitted	(ignored)	No labels. All cells in the input range are data.

Related Function

MCOVAR Returns the covariance between two or more data sets

MCOVAR

Returns a covariance matrix that measures the covariance between two or more data sets.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

MCOVAR(**inprng**, outrng, grouped, labels)

MCOVAR?(inprng, outrng, grouped, labels)

Inprng is the input range.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Grouped is a text character that indicates whether the data in the input range is organized by row or column.

- If grouped is "C" or omitted, then the data is organized by column.
- If grouped is "R", then the data is organized by row.

Labels is a logical value that describes where the labels are located in the input range, as shown in the following table:

Labels	Grouped	Labels are in
TRUE	"C"	First row of the input range
TRUE	"R"	First column of the input range
FALSE or omitted	(ignored)	No labels. All cells in the input range are data.

Related Function

MCORREL Returns the correlation coefficient of two or more data sets that are scaled to be independent of the unit of measurement

MENU.EDITOR

This function should not be used in Microsoft Excel 97 or later because the Menu Editor is only available in Microsoft Excel 95 and Microsoft Excel version 5.0.

MERGE.STYLES

Equivalent to clicking the Merge button in the Style dialog box, which appears when you click the Style command on the Format menu. Merges all the styles from another workbook into the active workbook. Use MERGE.STYLES when you want to import styles from another sheet in another workbook.

Syntax

MERGE.STYLES(document_text)

Document_text is the name of a sheet in a workbook from which you want to merge styles into the active workbook.

Remarks

- If any styles from the workbook being merged have the same name as styles in the active workbook, a dialog box appears asking if you want to replace the existing definitions of the styles with the "merged" definitions of the styles. If you click the Yes button, all the definitions are replaced; if you click the No button, all the original definitions in the active workbook are retained.
- When you move a sheet with styles to another workbook with styles, any styles with identical names but conflicting definitions have the sheet name added to the style name.

Related Functions

DEFINE.STYLE	Creates or changes a cell style
DELETE.STYLE	Deletes a cell style

MESSAGE

Displays and removes messages in the message area of the status bar. MESSAGE is useful for displaying text that doesn't need a response, such as descriptions of commands in user-defined menus.

Syntax

MESSAGE(**logical**, text)

Logical is a logical value specifying whether to display or remove a message.

- If logical is TRUE, Microsoft Excel displays text in the message area of the status bar.
- If logical is FALSE, Microsoft Excel removes any messages, and the status bar is returned to normal (that is, command help messages are displayed).

Text is the message you want to display in the status bar. If text is "" (empty text), Microsoft Excel removes any messages currently displayed in the status bar.

Remarks

- Only one message can be displayed in the status bar at a time. Messages are always displayed in the same place.
- MESSAGE works the same way whether the status bar is displayed or not. You can, for example, use MESSAGE while the status bar isn't displayed. As soon as you display the status bar, you see your message.
- If you display any message (even empty text) and don't remove it with MESSAGE(FALSE), that message is displayed until you quit Microsoft Excel.
- You can also use the ALERT function to get the user's attention; however, this interrupts the macro and requires the user's intervention before the macro can continue.

Example

The following lines in a macro display a message warning that you must wait for a moment while the macro calls a subroutine.

```
MESSAGE(TRUE, "One moment please...")
```

Related Functions

ALERT	Displays a dialog box and a message
BEEP	Sounds a tone

MOVE

Equivalent to moving a window by dragging its title bar in Microsoft Excel version 3.0 or earlier. MOVE is also equivalent to choosing the Move command from the Control menu in Microsoft Windows. This function is included only for macro compatibility and will be converted to WINDOW.MOVE when you open older macro sheets. For more information, see WINDOW.MOVE.

Syntax

MOVE(x_pos, y_pos, window_text)

MOVE?(x_pos, y_pos, window_text)

Related Functions

WINDOW.MOVE	Sizes a window
WINDOW.SIZE	Moves a window

MOVEAVG

Projects values in a forecast period, based on the average value of the variable over a specific number of preceding periods.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

MOVEAVG(**inprng**, outrng, interval, stderrs, chart, labels)

MOVEAVG?(inprng, outrng, interval, stderrs, chart, labels)

Inprng is the input range.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Interval is the number of values to include in the moving average. If omitted, interval is 3.

Stderrs is a logical value.

- If stderrs is TRUE, standard error values are included in the output table.
- If stderrs is FALSE or omitted, standard errors are not included in the output table.

Chart is a logical value.

- If chart is TRUE, then MOVEAVG generates a chart for the actual and forecast values.
- If chart is FALSE or omitted, the chart is not generated.

Labels is a logical value.

- If labels is TRUE, then the first row or column of inprng contains labels.
- If labels is FALSE or omitted, all cells in inprng are considered data. Microsoft Excel generates appropriate data labels for the output table.

MOVE.TOOL

Moves or copies a button from one toolbar to another.

Syntax

MOVE.TOOL(**from_bar_id, from_bar_position**, to_bar_id, to_bar_position, copy, width)

From_bar_id specifies the number or name of a toolbar from which you want to move or copy the button. For detailed information, see the description of bar_id in ADD.TOOL.

From_bar_position specifies the current position of the button within the toolbar. From_bar_position starts with 1 at the left side (if horizontal) or at the top (if vertical).

To_bar_id specifies the number or name of a toolbar to which you want to move or paste the button. For detailed information, see the description of bar_id in ADD.TOOL. To_bar_id is optional if you are moving a button within the same toolbar.

To_bar_position specifies where you want to move or paste the button within the toolbar. To_bar_position starts with 1 at the left side (if horizontal) or at the top (if vertical). To_bar_position is optional if you are only adjusting the width of a drop-down list.

Copy is a logical value specifying whether to copy the button. If copy is TRUE, the button is copied; if FALSE or omitted, the button is moved.

Width is the width, measured in points, of a drop-down list. If the button you are moving is not a drop-down list, width is ignored.

Related Functions

ADD.TOOL	Adds one or more buttons to a toolbar
COPY.TOOL	Copies a button face to the Clipboard
GET.TOOL	Returns information about a button or buttons on a toolbar

NAMES

Returns, as a horizontal array of text, the specified names defined in the specified workbook. The returned array lists the names in alphabetic order. Use NAMES instead of LIST.NAMES when you want to return the names to the macro sheet instead of to the active worksheet.

Syntax

NAMES(document_text, type_num, match_text)

Document_text is text that specifies the workbook whose names you want returned. If document_text is omitted, it is assumed to be the active workbook.

Type_num is a number from 1 to 3 that specifies whether to include hidden names in the returned array.

If type_num is	NAMES returns
1 or omitted	Normal names only
2	Hidden names only
3	All names

Match_text is text that specifies the names you want returned and can include wildcard characters. If match_text is omitted, all names are returned.

Remarks

- Hidden names are defined using the DEFINE.NAME macro function and do not appear in the Paste Name, Define Name, or Go To dialog boxes.

- NAMES returns a horizontal array, so you will normally enter this function as an array in several horizontal cells or define a name to refer to the array that NAMES returns. If you want the names in a vertical array instead, use the TRANSPOSE function.

- You can use the COLUMNS function to count the number of entries in the horizontal array.

Example

The following macro formula returns all names on the active workbook starting with the letter P.

```
NAMES(, 3, "P*")
```

Related Functions

DEFINE.NAME   Defines a name on the active worksheet or macro sheet

DELETE.NAME   Deletes a name

GET.DEF   Returns a name matching a definition

GET.NAME   Returns the definition of a name

LIST.NAMES   Lists names and their associated information

SET.NAME   Defines a name as a value

NEW

Equivalent to clicking the New command on the File menu. Creates a new Microsoft Excel workbook or opens a template.

Syntax

NEW(type_num, xy_series, add_logical)

NEW?(type_num, xy_series, add_logical)

Type_num   specifies the type of workbook to create, as shown in the following table. Type_num is most often 5 or quoted text; other values are mainly for compatibility with Microsoft Excel version 4.0.

Type_num	**Workbook**
Omitted	New workbook with a single worksheet of the same type as the active worksheet
1	New workbook with one worksheet
2	New workbook with one chart based on the current selection

3	New workbook with one macro sheet
4	New workbook with one international macro sheet
5	New workbook with 16 worksheets or based on the default workbook
6	New workbook with one Visual Basic module
7	New workbook with one dialog sheet
Quoted text	Template.

Xy_series is a number from 0 to 3 that specifies how data is arranged in a chart.

Xy_series	Result
0	Displays a dialog box if the selection is ambiguous.
1 or omitted	The first row/column is the first data series.
2	The first row/column contains the category (x) axis labels.
3	The first row/column contains the x-values; the created chart is an xy (scatter) chart.

Add_logical specifies whether or not to add the sheet type to the open workbook. If add_logical is TRUE, the sheet type is inserted before the current sheet; if FALSE or omitted, it is not inserted. This argument is for compatibility with Microsoft Excel version 4.0.

Add_logical is ignored if type_num is 5.

Remarks

You can also use NEW to create new sheets from templates that exist in the startup directory or folder, using for type_num the text that appears in the File New list box. To create new sheets from any template that is not in the start-up directory, use the OPEN function.

Related Functions

NEW.WINDOW Creates a new window for an existing worksheet or macro sheet

OPEN Opens a workbook

NEW.WINDOW

Equivalent to clicking the New Window command on the Window menu. Creates a new window for the active workbook.

Syntax

NEW.WINDOW( )

After you use NEW.WINDOW, use the WINDOW.MOVE, WINDOW.SIZE, and ARRANGE.ALL functions to size and position the new window.

Related Functions

ARRANGE.ALL   Arranges all displayed windows to fill the workspace and synchronizes windows for scrolling

WINDOW.MOVE   Moves a window

WINDOW.SIZE   Changes the size of a window

NEXT

Ends a FOR-NEXT, FOR.CELL-NEXT, or WHILE-NEXT loop and continues carrying out the current macro with the formula that follows the NEXT function.

Syntax

NEXT( )

Related Functions

FOR   Starts a FOR-NEXT loop

FOR.CELL   Starts a FOR.CELL-NEXT loop

WHILE   Starts a WHILE-NEXT loop

NOTE

Equivalent to choosing the Comment command from the Insert menu. Creates a comment or replaces text characters in a comment.

Syntax

NOTE(add_text, cell_ref, start_char, num_chars)

NOTE?( )

Add_text   is text of up to 255 characters you want to add to a comment. Add_text must be enclosed in quotation marks.

- If add_text is omitted, it is assumed to be "" (empty text).

Cell_ref   is the cell to which you want to add the comment text. If cell_ref is omitted, add_text is added to the active cell's comment.

Start_char is the number of the character at which you want add_text to be added. If start_char is omitted, it is assumed to be 1. If there is no existing comment, start_char is ignored.

Num_chars is the number of characters that you want to replace in the comment. If num_chars is omitted, it is assumed to be equal to the length of the comment.

Remarks

- NOTE returns the number of the last character entered in the comment. This is useful if you want to know how many characters are in the text string.
- The dialog-box form of this function, NOTE?, takes no arguments.
- `NOTE()` deletes the comment attached to the active cell.

To find out if a cell has a comment attached to it, use GET.CELL.

Related Function

GET.NOTE Returns characters from a comment

OBJECT.PROPERTIES

Determines how the selected object or objects are attached to the cells beneath them and whether they are printed. The way an object is attached to the cells beneath it affects how the object is moved or sized whenever you move or size the cells.

OBJECT.PROPERTIES(placement_type, print_object)

OBJECT.PROPERTIES?(placement_type, print_object)

Placement_type is a number from 1 to 3 specifying how to attach the selected object or objects. If placement_type is omitted, the current status is unchanged.

If placement_type is	The selected object is
1	Moved and sized with cells.
2	Moved but not sized with cells.
3	Free-floating—it is not affected by moving and sizing cells.

Print_object is a logical value specifying whether to print the selected object or objects. If TRUE or omitted, the objects are printed; if FALSE, they are not printed.

Remarks

If an object is not selected, OBJECT.PROPERTIES interrupts the macro and returns the #VALUE! error value.

Related Functions

CREATE.OBJECT   Creates an object

FORMAT.MOVE   Moves the selected object

FORMAT.SIZE   Changes the size of the selected object

OBJECT.PROTECTION

Changes the protection status of the selected object.

Syntax

OBJECT.PROTECTION(locked, lock_text)

OBJECT.PROTECTION?(locked, lock_text)

Locked   is a logical value that determines whether the selected object is locked or unlocked. If locked is TRUE, Microsoft Excel locks the object; if FALSE, Microsoft Excel unlocks the object.

Lock_text   is a logical value that determines whether text in a text box or button can be changed. Lock_text applies only if the object is a text box, button, or worksheet control. If lock_text is TRUE or omitted, text cannot be changed; if FALSE, text can be changed.

Remarks

- You cannot lock or unlock an individual object with OBJECT.PROTECTION when protection is selected for objects in the Protect Sheet dialog box.
- If an object is not selected, the function returns the #VALUE! error value and halts the macro.
- In order for an object to be protected, you must use the PROTECT.DOCUMENT(, , , TRUE) function after changing the object's status with OBJECT.PROTECTION.

Related Functions

PROTECT.DOCUMENT   Controls protection for the active worksheet

WORKBOOK.PROTECT   Controls protection for the active workbook

ON.DATA

Runs a specified macro when another application sends data to a particular workbook via dynamic data exchange (DDE) or via Publish and Subscribe on the Macintosh. Workbook links to other applications are called remote references.

Syntax

ON.DATA(document_text, macro_text)

Important   Microsoft Excel for the Macintosh requires system software version 7.0 or later for this function.

Document_text   is the name of the sheet to which remote data will be sent or the name of the source of the remote data.

- If document_text is the name of the remote data source, it must be in the form app|topic!item. You can use the form app|topic to include all items for a particular topic, or app| to specify an app alone, but you must include the | to indicate that you are specifying the name of a data source.
- If document_text is omitted, the macro specified by macro_text is run whenever remote data is sent to any sheet not already assigned to another ON.DATA function.
- In Microsoft Excel for the Macintosh, document_text can also be the name of a published edition file. Unless the file is in the current folder, document_text must include the complete path.

Macro_text is the name of, or an R1C1-style reference to, a macro that you want to run when data comes into the workbook or from the source specified by document_text. The name or reference must be in text form.

- If macro_text is omitted, the ON.DATA function is turned off for the specified workbook or source.

Remarks

- ON.DATA remains in effect until you either clear it or quit Microsoft Excel. You can clear ON.DATA by specifying document_text and omitting the macro_text argument.
- If the macro sheet containing macro_text is closed when data is sent to document_text, an error is returned.
- If the incoming data causes recalculation, Microsoft Excel first runs the macro macro_text and then performs the recalculation.

Examples

In Microsoft Excel for Windows, the following macro formula runs the macro AddOrders when data is sent to the sheet New in the workbook ORDERSDB.XLS:

```
ON.DATA("[ORDERSDB.XLS]New", "AddOrders")
```

In Microsoft Excel for the Macintosh, the following macro formula runs the macro beginning at cell R2C3 when data is sent to the sheet North in the workbook SALES DATABASE:

```
ON.DATA("[SALES DATABASE]North", "R2C3")
```

Related Functions

ERROR Specifies what action to take if an error is encountered while a macro is running

INITIATE Opens a channel to another application

ON.ENTRY Runs a macro when data is entered

ON.RECALC Runs a macro when a workbook is recalculated

ON.DOUBLECLICK

Runs a macro when you double-click any cell or object on the specified sheet or macro sheet or double-click any item on the specified chart.

Syntax

ON.DOUBLECLICK(sheet_text, macro_text)

Sheet_text is a text value specifying the name of a sheet in a workbook. If sheet_text is omitted, the macro is run whenever you double-click any sheet not specified by a previous ON.DOUBLECLICK formula. Sheet_text must be in the form "[book1]sheet1".

Macro_text is the name of, or an R1C1-style reference to, a macro you want to run when you double-click the sheet specified by sheet_text. The name or reference must be in text form. If macro_text is omitted, double-clicking reverts to its normal behavior, and any macros assigned by previous ON.DOUBLECLICK functions are turned off.

Remarks

- ON.DOUBLECLICK overrides Microsoft Excel's normal double-click behavior, such as displaying cell notes, displaying the Patterns dialog box, or allowing editing cell text directly in cells.
- To determine what cell, object, or chart item has been double-clicked, use a CALLER function in the macro specified by macro_text.
- ON.DOUBLECLICK does not affect objects to which ASSIGN.TO.OBJECT macros have already been assigned. Use ON.DOUBLECLICK (TRUE) to make Microsoft Excel carry out the action that would normally occur if you double-click on the current selection.

Related Functions

ASSIGN.TO.OBJECT Assigns a macro to an object

ON.WINDOW Runs a macro when you switch to a window

ON.ENTRY

Runs a macro when you enter data into any cell on the specified sheet.

Syntax

ON.ENTRY(sheet_text, macro_text)

Sheet_text is a text value specifying the name of a sheet in a workbook. If sheet_text is omitted, the macro is run whenever you enter data into any sheet or macro sheet.

Macro_text is the name of, or an R1C1-style reference to, a macro you want to run when you enter data into the sheet specified by sheet_text. The name or reference must be in text form. If macro_text is omitted, entering data reverts to its normal behavior, and any macros assigned by previous ON.ENTRY functions are turned off.

Remarks

- The macro is run only when you enter data in a cell, not when you use edit commands or macro functions.

- To determine what cell had data entered into it, use a CALLER function in the macro specified by macro_text.

Related Functions

ENTER.DATA Turns Data Entry mode on or off

ON.RECALC Runs a macro when a workbook is recalculated

ON.KEY

Runs a specified macro when a particular key or key combination is pressed.

Syntax

ON.KEY(**key_text**, macro_text)

Key_text can specify any single key, or any key combined with ALT, CTRL, or SHIFT, or any combination of those keys (in Microsoft Excel for Windows) or COMMAND, CTRL, OPTION, or SHIFT or any combination of those keys (in Microsoft Excel for the Macintosh). Each key is represented by one or more characters, such as "a" for the character a, or "{ENTER}" for the ENTER key.

To specify characters that aren't displayed when you press the key, such as ENTER or TAB, use the codes shown in the following table. Each code in the table represents one key on the keyboard.

Key	**Code**
BACKSPACE	"{BACKSPACE}" or "{BS}"
BREAK	"{BREAK}"
CAPS LOCK	"{CAPSLOCK}"
CLEAR	"{CLEAR}"
DELETE or DEL	"{DELETE}" or "{DEL}"
DOWN	"{DOWN}"
END	"{END}"
ENTER (numeric keypad)	"{ENTER}"

ENTER	"~" (tilde)
ESC	"{ESCAPE} or {ESC}"
HELP	"{HELP}"
HOME	"{HOME}"
INS	"{INSERT}"
LEFT	"{LEFT}"
NUM LOCK	"{NUMLOCK}"
PAGE DOWN	"{PGDN}"
PAGE UP	"{PGUP}"
RETURN	"{RETURN}"
RIGHT	"{RIGHT}"
SCROLL LOCK	"{SCROLLLOCK}"
TAB	"{TAB}"
UP	"{UP}"
F1 through F15	"{F1}" through "{F15}"

In Microsoft Excel for Windows, you can also specify keys combined with SHIFT and/or CTRL and/or ALT. In Microsoft Excel for the Macintosh, you can also specify keys combined with SHIFT and/or CTRL and/or OPTION and/or COMMAND. To specify a key combined with another key or keys, use the following table.

To combine with	Precede the key code by
SHIFT	"+" (plus sign)
CTRL	"^" (caret)
ALT or OPTION	"%" (percent sign)
COMMAND	"*" (asterisk)

To assign a macro to one of the special characters (+, ^, %, and so on), enclose the character in brackets. For example, ON.KEY("^{+}", "InsertItem") assigns a macro named InsertItem to the key sequence CTRL+PLUS SIGN.

Macro_text is the name of a macro that you want to run when key_text is pressed. The reference must be in text form.

- If macro_text is "" (empty text), nothing happens when key_text is pressed. This form of ON.KEY disables the normal meaning of keystrokes in Microsoft Excel.
- If macro_text is omitted, key_text reverts to its normal meaning in Microsoft Excel, and any special key assignments made with previous ON.KEY functions are cleared.

Remarks

- ON.KEY remains in effect until you clear it or quit Microsoft Excel. You can clear ON.KEY by specifying key_text and omitting the macro_text argument.
- If the macro sheet containing macro_text is closed when you press key_text, an error is returned.
- If another macro is running when you press key_text, macro_text will not run.

Examples

Suppose you wanted the key combination SHIFT+CTRL+RIGHT to run the macro Print. You use the following macro formula:

```
ON.KEY("+^{RIGHT}", "Print")
```

To return SHIFT+CTRL+RIGHT to its normal meaning, you would use the following macro formula:

```
ON.KEY("+^{RIGHT}")
```

To disable SHIFT+CTRL+RIGHT altogether, you would use the following macro formula:

```
ON.KEY("+^{RIGHT}", "")
```

Related Functions

CANCEL.KEY	Disables macro interruption
ERROR	Specifies what action to take if an error is encountered while a macro is running
SEND.KEYS	Sends a key sequence to an application

ON.RECALC

Runs a macro when a specific sheet is recalculated. Use ON.RECALC to perform an operation on a sheet each time the sheet is recalculated, such as checking that a certain condition is still being met.

Syntax

ON.RECALC(sheet_text, macro_text)

Sheet_text is the name of a sheet, given as text. If sheet_text is omitted, the macro is run whenever any open sheet not specified by a previous ON.RECALC formula is recalculated. Only one ON.RECALC formula can be run for each recalculation.

Macro_text is the name of, or an R1C1-style reference to, a macro you want to run when the sheet specified by sheet_text is recalculated. The name or reference must be in text form. The macro will be run each time the sheet is recalculated until ON.RECALC is cleared. If macro_text is omitted, recalculating reverts to its normal behavior, and any macros assigned by previous ON.RECALC functions are turned off.

Remarks

A macro specified to be run by ON.RECALC is not run by actions taken by other macros. For example, a macro specified by ON.RECALC will not be run after the CALCULATE.NOW function is carried out, but will be run if you change data in a sheet set to calculate automatically or choose the Calc Now button in the Calculation panel of the Options dialog box, which appears when you choose the Options command from the Tools menu.

Examples

In Microsoft Excel for Windows, the following macro formula specifies that the macro Printer on the macro sheet AUTOREPT.XLS be run when the worksheet named REPORT.XLS is recalculated:

```
ON.RECALC("REPORT.XLS", "AUTOREPT.XLS!Printer")
```

In Microsoft Excel for the Macintosh, the following macro formula turns off ON.RECALC for the workbook named SALES:

```
ON.RECALC("SALES")
```

Related Functions

CALCULATE.DOCUMENT	Calculates the active sheet only
CALCULATE.NOW	Calculates all open workbooks immediately

CALCULATION Controls calculation settings

ON.ENTRY Runs a macro when data is entered

ON.SHEET

Starts a macro whenever the specified sheet is activated from another sheet.

Syntax

ON.SHEET(sheet_text, macro_text, activate_logical)

Sheet_text is the name of the sheet that triggers a macro when it is activated, in the form "[Book1]Sheet1". If omitted, then when any sheet in any book is activated, macro_text will run.

Macro_text is the name of the macro to run when the specified sheet is activated. If omitted, then the triggering of a macro on the specified sheet is cancelled.

Activate_logical is a logical value that specifies if the macro is run when the sheet is activated or deactivated. If TRUE or omitted, the macro is run when the sheet is activated. If FALSE, the macro is run when the sheet is deactivated.

Examples

`ON.SHEET("[STORE.XLS]Sheet1","WeeklyCalc")` will run the macro "WeeklyCalc" when "[STORE.XLS]Sheet1" is activated.

`ON.SHEET(,"WeeklyCalc")` runs "WeeklyCalc" when any sheet in the book is activated.

`ON.SHEET("[STORE.XLS]Sheet1")` cancels the triggering of "WeeklyCalc" when Sheet1 in the book STORE.XLS is activated.

`ON.SHEET("[STORE.XLS]","WeeklyCalc")` runs "WeeklyCalc" when any sheet in the book STORE.XLS is activated

Related Function

ON.WINDOW Runs a specified macro when you switch to a particular window.

ON.TIME

Runs a macro at a specified time. Use ON.TIME to run a macro at a specific time of day or after a specified period has passed.

Syntax

ON.TIME(**time, macro_text**, tolerance, insert_logical)

Time is the time and date, given as a serial number, at which the macro is to be run. If time does not include a date (that is, if time is a serial number less than 1), the macro is run the next time this time occurs.

Macro_text is the name of, or an R1C1-style reference to, a macro to run at the specified time and every subsequent day at that time.

Tolerance is the time and date, given as a serial number, that you are willing to wait until and still have the macro run. For example, if Microsoft Excel is not in Ready, Copy, Cut, or Find mode at time, because another macro is running, but is in Ready mode 15 seconds later, and tolerance is set to time plus 30 seconds, the macro specified by macro_text will run. If Microsoft Excel was not in Ready mode within 30 seconds, the macro would not run. If tolerance is omitted, it is assumed to be infinite.

Insert_logical is a logical value specifying whether you want every day macro_text to run at time. Use insert_logical when you want to clear a previously set ON.TIME formula. If insert_logical is TRUE or omitted, the macro specified by macro_text is carried out at time. If insert_logical is FALSE and macro_text is not set to run at time, ON.TIME returns the #VALUE error value.

Examples

The following macro formula runs a macro called Test at 5:00:00 P.M. every day when Microsoft Excel is in Ready mode:

```
ON.TIME("5:00:00 PM", "Test")
```

The following macro formula runs a macro called Test 5 seconds after the formula is evaluated:

```
ON.TIME(NOW()+"00:00:05", "Test")
```

The following macro formula runs a macro called Test 10 seconds after the formula is evaluated. If Microsoft Excel is not in Ready mode at that time (because it is in Edit mode, for example), the tolerance argument specifies 5 seconds of additional time to wait to run the macro. If Microsoft Excel is still not in Ready mode at that time, macro_text is not run.

```
ON.TIME(NOW()+"00:00:10", "Test", NOW()+"00:00:15")
```

ON.WINDOW

Runs a specified macro when you switch to a particular window.

Syntax

ON.WINDOW(window_text, macro_text)

Window_text is the name of a window in the form of text. If window_text is omitted, ON.WINDOW starts the macro whenever you switch to any window, except for windows that are named in other ON.WINDOW statements.

Macro_text is the name of, or an R1C1-style reference to, a macro to run when you switch to window_text. If macro_text is omitted, switching to window_text no longer runs the previously specified macro.

Remarks

- A macro specified to be run by ON.WINDOW is not run when other macros switch to the window or when a command to switch to a window is received through a DDE channel. Instead, ON.WINDOW responds to a user's actions, such as clicking a window with the mouse, clicking the Copy command on the Edit menu, and so on.

- If a sheet or macro sheet has an Auto_Activate or Auto_Deactivate macro defined for it, those macros will be run after the macro specified by ON.WINDOW.

Examples

In Microsoft Excel for Windows, the following macro formula runs the macro beginning at cell R1C2 when you switch to the window MAIN.XLS:

```
ON.WINDOW("MAIN.XLS", "R1C2")
```

The following macro formula stops the macro from running when you switch to MAIN.XLS:

```
ON.WINDOW("MAIN.XLS")
```

In Microsoft Excel for the Macintosh, the following macro formula runs the macro named ShowAlert when you switch to the window MAIN WINDOW:

```
ON.WINDOW("MAIN WINDOW", "ShowAlert")
```

The following macro formula stops the macro from running when you switch to MAIN WINDOW:

```
ON.WINDOW("MAIN WINDOW")
```

Related Functions

GET.WINDOW	Returns information about a window
ON.KEY	Runs a macro when a specified key is pressed
ON.SHEET	Triggers a macro whenever the specified sheet is activated from another sheet
WINDOWS	Returns the names of all open windows

OPEN

Equivalent to clicking the Open command on the File menu. Opens an existing workbook.

Syntax

OPEN(file_text, update_links, read_only, format, prot_pwd, write_res_pwd, ignore_rorec, file_origin, custom_delimit, add_logical, editable, file_access, notify_logical, converter)

OPEN?(file_text, update_links, read_only, format, prot_pwd, write_res_pwd, ignore_rorec, file_origin, custom_delimit, add_logical, editable, file_access, notify_logical, converter)

File_text is the name, as text, of the workbook you want to open. File_text can include a drive and path, and can be a network pathname. In the dialog-box form in Microsoft Excel for Windows, file_text can include an asterisk (*) to represent any sequence of characters and a question mark (?) to represent any single character.

Update_links specifies whether and how to update external and remote references. If update_links is omitted, Microsoft Excel displays a message asking if you want to update links.

If update_links is	Then Microsoft Excel
0	Updates neither external nor remote references
1	Updates external references only
2	Updates remote references only
3	Updates external and remote references

Note When you are opening a file in WKS, WK1, or WK3 format, the update_links argument specifies whether Microsoft Excel generates charts from any graphs attached to the WKS, WK1, or WK3 file.

If update_links is	Charts are
0	Not created
2	Created

Read_only corresponds to the Read Only check box in the Open dialog box. If read_only is TRUE, the workbook can be modified but changes cannot be saved; if FALSE or omitted, changes to the workbook can be saved.

Format specifies what character to use as a delimiter when opening text files. If format is omitted, Microsoft Excel uses the current delimiter setting.

If format is	Values are separated by
1	Tabs
2	Commas
3	Spaces
4	Semicolons

5	Nothing
6	Custom characters

Prot_pwd is the password, as text, required to unprotect a protected file. If prot_pwd is omitted and file_text requires a password, the Password dialog box is displayed. Passwords are case-sensitive. Passwords are not recorded when you open a workbook and supply the password with the macro recorder on.

Write_res_pwd is the password, as text, required to open a read-only file with full write privileges. If write_res_pwd is omitted and file_text requires a password, the Password dialog box is displayed.

Ignore_rorec is a logical value that controls whether the read-only recommended message is displayed. If ignore_rorec is TRUE, Microsoft Excel prevents display of the message; if FALSE or omitted, and if read_only is also FALSE or omitted, Microsoft Excel displays the alert when opening a read-only recommended workbook.

File_origin is a number specifying whether a text file originated on the Macintosh or in Windows.

File_origin	**Original operating environment**
1	Macintosh
2	Windows (ANSI)
3	MS-DOS (PC-8)

OmittedCurrent operating environment

Custom_delimit is the character you want to use as a custom delimiter when opening text files.

- Custom_delimit is text or a reference or formula that returns text, such as CHAR(124).
- Custom_delimit is required if format is 6; it is ignored if format is not 6.
- Only the first character in custom_delimit is used.

Add_logical is a logical value that specifies whether or not to add file_text to the open workbook. If add_logical is TRUE, the document is added; if FALSE or omitted, it is not added. This argument is for compatibility with workbooks from Microsoft Excel version 4.0.

Editable is a logical value that corresponds to opening a file (such as a template) while holding down SHIFT key. If TRUE, editable is the equivalent to holding down the SHIFT key while clicking the OK button in the Open dialog box. If FALSE or omitted, this argument is ignored.

File_access is a number specifying how the file is to be accessed. If the file is being opened for the first time, this argument is ignored. If the file is already opened, this argument determines how to change the user's access permissions for the file.

File Access	How Accessed
1	Revert to saved copy
2	Change to read/write access
3	Change to read only access

Notify_logical is a logical value that specifies whether the user should be notified when the shared workbook is available to be opened across a network. If TRUE, the user will be notified when the workbook is available to be opened. If FALSE or omitted, the user will not be notified when the file available to be opened.

Converter is a number corresponding to the file converter to use to open the file. Normally, Microsoft Excel automatically determines which file converter to use; therefore, this argument can usually be excluded. If you want to be certain, however, that a specific manually installed converter be used, then include this argument. Use GET.WORKSPACE(62) to determine which numbers corresponds to all of the installed converters.

Related Functions

CLOSE Closes the active window

FCLOSE Closes a text file

FOPEN Opens a file with the type of permission specified

OPEN.LINKS Opens specified supporting workbooks

OPEN.DIALOG

Displays the standard Microsoft Excel File Open dialog box with the specified file filters. When the user presses the button specified by button_text, the filename the user picks or types in will be returned.

Syntax

OPEN.DIALOG(file_filter, button_text, title, filter_index)

File_filter is the file filtering criteria to use, as text. For Microsoft Excel for Windows, file_filter consists of two parts, a descriptive phrase denoting the file type followed by a comma and then the MS-DOS wildcard file filter specification, as in "Text Files (*.TXT), *.TXT, Add-in Files (*.XLA), *.XLA". Groups of filter specifications are also separated by commas. Each separate pair is listed in the file type drop-down list box. File_filter can include an asterisk (*) to represent any sequence of characters and a question mark (?)

to represent any single character. For Microsoft Excel for the Macintosh, file_filter consists of file type codes separated by commas, as in "TEXT,XLA ,XLS4". Spaces are significant and should not be inserted before or after the comma separators unless they are part of the file type code.

Button_text is the text used for the Open button in the dialog. If omitted, "Open" will be used. Button_text is ignored on Microsoft Excel for Windows.

Title specifies the dialog window title. If omitted, "File Open" will be used as the default.

Filter_index specifies the index number of the default file filtering criteria from 1 to the number of filters specified in file_filter. If omitted or greater than the number of filters present, 1 will be used as the starting index number. The argument is ignored on Microsoft Excel for the Macintosh.

Remarks

- To use multiple MS-DOS wildcard expressions within file_filter for a single file filter type, separate the wildcard expressions with semicolons, as in "VB Files (*.bas; *.txt), *.bas;*.txt"
- If file_filter is omitted, "ALL Files (*.*), *.*" will be used as the default in Microsoft Excel for Windows. The default for Microsoft Excel for the Macintosh is all file types.
- If the user cancels the dialog box, FALSE is returned.

Examples

`OPEN.DIALOG("Text Files (*.TXT), *.TXT, Add-in Files (*.XLA), *.XLA,ALL FILES (*.*), *.*","FILE OPENER")` opens a file open dialog box titled "FILE OPENER" with three file filter criteria in the drop-down list box.

Related Function

SAVE.DIALOG Displays the standard Microsoft Excel File Save As dialog box and gets a file name from the user

OPEN.LINKS

Equivalent to clicking the Links command on the Edit menu. Use OPEN.LINKS with the LINKS function to open workbooks linked to a particular sheet.

Syntax

OPEN.LINKS(**document_text1**, document_text2, ..., read_only, type_of_link)

OPEN.LINKS?(document_text1, document_text2, ..., read_only, type_of_link)

Document_text1, document_text2, are 1 to 12 arguments that are the names of supporting documents in the form of text, or arrays or references that contain text.

Read_only is a logical value corresponding to the read/write status of the linked worksheet. If read_only is TRUE, the sheet can be modified but changes cannot be saved; if FALSE or omitted, changes to the sheet can be saved. Read_only applies only to Microsoft Excel, WKS, and SYLK documents.

Type_of_link is a number from 1 to 6 that specifies what type of link you want to get information about.

Type_of_link	Link document type
1	Microsoft Excel link
2	DDE link
3	Reserved
4	Not applicable
5	Subscriber
6	Publisher

Remarks

You can generate an array of the names of linked workbooks with the LINKS function.

Related Functions

CHANGE.LINK   Changes supporting workbook links

GET.LINK.INFO   Returns information about a link

LINKS   Returns the name of all linked workbooks

UPDATE.LINK   Updates a link to another document

OPEN.MAIL

Equivalent to clicking the Open Mail command on the Mail submenu on File menu.

Note   This function is available for only Microsoft Excel for the Macintosh and Microsoft Mail version 2.0 or later.

Syntax

OPEN.MAIL(subject, comments)

OPEN.MAIL?(subject, comments)

Subject   is the subject of the message containing a file that Microsoft Excel can open.

- For each message whose subject matches the subject argument and contains a file that Microsoft Excel can open, the file is opened in Microsoft Excel; if the message has no unread enclosures, it is deleted from the list of pending mail.

- If subject is omitted, then for all messages containing files that Microsoft Excel can open, the files are opened; each message that has no unread enclosures is deleted from the list of pending mail.

Comments is a logical value that specifies whether comments associated with the Microsoft Excel files are displayed. If comments is TRUE, Microsoft Excel displays the comments; if FALSE, comments are not displayed. If omitted, the current setting is not changed.

Tips

- If you use consistent subjects in your Microsoft Mail messages, you can easily create a macro that always opens mail messages with certain files attached. For example, an OPEN.MAIL formula with subject specified as "Weekly Report" would open the Microsoft Excel file attached to the message containing that subject each week.
- In OPEN.MAIL?, the dialog-box form of the function, the currently running macro pauses while the Microsoft Mail documents window is displayed. The macro resumes after you close the Microsoft Mail documents window.

Related Function

SEND.MAIL Sends the active workbook

OPEN.TEXT

Equivalent to using the Text Import Wizard to open a text file in Microsoft Excel.

Syntax

OPEN.TEXT(file_name, file_origin, start_row, file_type, text_qualifier, consecutive_delim, tab, semicolon, comma, space, other, other_char, {field_info1; field_info2;...})

File_name is the full pathname of the text file you want to open.

File_origin specifies the operating environment the text file was created in.

File_origin	Operating system
1	Macintosh
2	Windows (ANSI)
3	MS DOS (PC-8)
Omitted	Current operating environment

Start_row is a number greater than or equal to one, specifying the row in the text file where you want to start importing into Microsoft Excel. This number should be less than the number of lines in the text file.

File_type specifies the type of delimited text file to import:

File_type	Type of file
1 or omitted	Delimited
2	Fixed width

Text_qualifier indicates the character-enclosing text fields in the text file:

Text_qualifier value	Qualifier
1 or "	" (double quotation mark)
2 or '	' (single quotation mark)
3 or {None}	No text qualifier

Consecutive_delim is a logical value corresponding to the Treat Consecutive Delimiters as One check box which, if TRUE, allows consecutive delimiters (such as ",,,") to be treated as a single delimiter. If FALSE, all consecutive delimiters are considered separate column breaks.

Tab, semicolon, comma, space are logical values corresponding to the check boxes in the Column Delimiters group. If an argument is TRUE, the check box is selected. These arguments apply only when the file_type argument is 1 or omitted (delimited file type).

Other is a logical value indicating a custom delimiter to use in opening the text file.

Other_char specifies the custom delimiter to use or FALSE if no custom delimiter is used.

Field_info is an array which consists of the following elements: "column number, data_format", if file_type is 1; or "start_pos, data_format" if file_type is 2.

Related Function

TEXT.TO.COLUMNS Parses text, as in a text file, into columns of data

OPTIONS.CALCULATION

Equivalent to clicking the Options command on the Tools menu, and clicking the Calculation tab in the Options dialog box. Sets various worksheet calculation settings.

Syntax

OPTIONS.CALCULATION(type_num, iter, max_num, max_change, update, precision, date_1904, calc_save, save_values)

OPTIONS.CALCULATION?(type_num, iter, max_num, max_change, update, precision, date_1904, calc_save, save_values)

Arguments correspond to check boxes and options in the Calculation tab in the Options dialog box. Arguments that correspond to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box.

Type_num   is a number from 1 to 3 indicating the type of calculation.

Type_num	**Type of calculation**
1	Automatic
2	Automatic except tables
3	Manual

Iter   corresponds to the Iteration check box. The default is FALSE.

Max_num   is the maximum number of iterations. The default is 100.

Max_change   is the maximum change of each iteration. The default is 0.001.

Update   corresponds to the Update Remote References check box. The default is TRUE.

Precision   corresponds to the Precision As Displayed check box. The default is FALSE.

Date_1904   corresponds to the 1904 Date System check box. The default is FALSE in Microsoft Excel for Windows and TRUE in Microsoft Excel for the Macintosh.

Calc_save   corresponds to the Recalculate Before Save check box. If calc_save is FALSE, the workbook is not recalculated before saving when in manual calculation mode. The default is TRUE.

Save_values   corresponds to the Save External Link Values check box. The default is TRUE.

Note   Microsoft Excel for Windows and Microsoft Excel for the Macintosh use different date systems as their default. For more information, see NOW.

OPTIONS.CHART

Equivalent to clicking the Options command on the Tools menu and then clicking the Chart Tab in the Options dialog box when a chart is activated for editing. Sets various chart settings.

Syntax

OPTIONS.CHART(Display_Blanks, Plot_Visible, Size_With_Window)

OPTIONS.CHART?(Display_Blanks, Plot_Visible, Size_With_Window)

Display_Blanks   is a number indicating how blank cells are plotted.

Number	Blanks are displayed as
1	Not plotted (gaps are shown)
2	Zero values
3	Interpolated

Plot_Visible is a logical value that if TRUE plots only visible data. If FALSE, all cells in the selection are plotted.

Size_With_Window is a logical value that if TRUE allows the chart to resize with window. If FALSE, chart will not size with window.

Remarks

If any of the arguments are omitted, then that setting is unchanged within the Options dialog box.

Related Functions

PREFERRED	Changes the format of the active chart
SET.PREFERRED	Changes the default chart format

OPTIONS.EDIT

Equivalent to clicking the Options command on the tools menu and then clicking the Edit tab in the Options dialog box. Sets various worksheet editing options.

Syntax

OPTIONS.EDIT(incell_edit, drag_drop, alert, entermove, fixed, decimals, copy_objects, update_links, move_direction, autocomplete, animations)

OPTIONS.EDIT?(incell_edit, drag_drop, alert, entermove, fixed, decimals, copy_objects, update_links, move_direction, autocomplete, animations)

Incell_edit is a logical value corresponding to the Edit Directly In Cell check box, which if TRUE allows In Cell Editing. If FALSE, editing directly in cells is not allowed. If omitted, the dialog box setting is not changed.

Drag_drop is a logical value corresponding to the Allow Cell Drag And Drop check box, which if TRUE allows drag and dropping on sheets. If FALSE, drag and drop is not allowed. If omitted, the dialog box setting is not changed.

Alert is a logical value corresponding to the Alert Before Overwriting Cells check box, which if TRUE displays an alert message warning you that cells containing values are about to be overwritten. If FALSE, an alert will not be displayed if your cells are about to overwritten. If omitted, the dialog box setting is not changed.

Entermove is a logical value corresponding to the Move Selection After Enter check box, which if TRUE moves the selection after the ENTER key is pressed. If FALSE, the selection is not moved. If omitted, the dialog box setting is not changed.

Fixed is a logical value corresponding to the Fixed Decimal check box, which if TRUE fixes the decimal place according to decimals. If FALSE, the decimal places are not fixed. If omitted, the dialog box setting is not changed.

Decimals is a number specifying the number of decimal places. Decimals is ignored if fixed is FALSE or omitted.

Copy_objects is a logical value corresponding to the Cut, Copy, And Sort Objects With Cells check box. If TRUE allows objects to be cut, copied and sorted with their cells. If FALSE, objects are not cut, copied or sorted with cells. If omitted, the dialog box setting is not changed.

Update_links is a logical value corresponding to the Ask To Update Automatic Links check box, which if TRUE will prompt the user when the workbook is opened that has links to other documents. If FALSE, the prompt will not be displayed. If omitted, the dialog box setting is not changed.

Move_direction is a number specifying the direction to move the selection when the ENTER key is pressed and Entermove is TRUE. Setting this number to 1 moves down one cell, 2 moves right one cell, 3 moves up on cell, and 4 moves down one cell..

Autocomplete is a logical value corresponding to the Enable AutoComplete For Cell Values check box, which, if TRUE, will enable the AutoComplete feature of Microsoft Excel 95 and later versions.

Animations is a logical value corresponding to the Provide Feedback With Animation check box, which if TRUE will enable the worksheet Animations feature of Microsoft Excel 95 and later versions. Deleted worksheet rows and columns will slowly disappear, and inserted worksheet rows and columns will slowly appear.

Related Function

OPTIONS.GENERAL Sets various general Microsoft Excel settings

OPTIONS.GENERAL

Equivalent to clicking the Options command on the Tools menu and then clicking the General tab from the Options dialog box. Sets various general Microsoft Excel settings.

Syntax

OPTIONS.GENERAL(R1C1_mode, dde_on, sum_info, tips, recent_files, old_menus, user_info, font_name, font_size, default_location, alternate_location, sheet_num, enable_under)

OPTIONS.GENERAL?(R1C1_mode, dde_on, sum_info, tips, recent_files, old_menus, user_info, font_name, font_size, default_location, alternate_location, sheet_num, enable_under)

Arguments correspond to option buttons, check boxes and text boxes on the General tab of the Options dialog box. Arguments corresponding to check boxes are logical values. If an argument is TRUE, the check box is selected; if FALSE, the check box is cleared; if omitted, the current setting is not changed.

R1C1_mode is a number specifying the reference style. Use 1 for A1 style references; 2 for R1C1 style references.

Dde_on is a logical value corresponding to the Ignore Other Applications check box, which if TRUE ignores DDE request from other applications. If FALSE, DDE requests from other applications are allowed to happen.

Sum_info is a logical value corresponding to the Prompt For Workbook Properties check box, which if TRUE displays the Summary tab of the Properties dialog box when a workbook is initially saved. If FALSE, the dialog box is not displayed.

Tips is a logical value corresponding to the Reset TipWizard check box in Microsoft Excel 95 or earlier versions, which, if TRUE, resets the TipWizard. In Microsoft Excel 97 or later, tips corresponds to the Reset My Tips button in the Office Assistant dialog box. If FALSE, the TipWizard will not be reset.

Recent_files is a logical value corresponding to the Recently Used File List check box, which if TRUE displays the four last opened files from the File menu. If FALSE, the file list will not be displayed.

Old_menus is a logical value corresponding to the Microsoft Excel 4.0 Menus check box in Microsoft Excel version 5.0, which if TRUE replaces the current menu bar with the Microsoft Excel 4.0 menu bar. If FALSE, the menu bar will not be replaced. This argument is for compatibility with Microsoft Excel version 5.0 only and is ignored in later versions. User_info corresponds to the Name text box, and is the name of the user of this copy of Microsoft Excel. By default it is the registered user, but can be changed to work on a network.

Font_name corresponds to the Standard Font text box, and is the name of the default font.

Font_size corresponds to the Size drop-down edit box, and is the size of the default font.

Default_location corresponds to the Default File Location text box, and is the default location that the File Open command displays. The Default is where Microsoft Excel is installed.

Alternate_location corresponds to the Alternate Startup File Location text box, and is the alternate startup directory.

Sheet_num corresponds to the Sheets In New Workbook spin control, and is the number of sheets in a new workbook. Default is 3. Can go up to 255.

Enable_under enables underlining of the menus. Used for only Microsoft Excel for the Macintosh. Ignored in Microsoft Excel for Windows.

Related Functions

OPTIONS.LISTS.DELETE Deletes a custom list

OPTIONS.LISTS.GET Returns contents of custom AutoFill lists

OPTIONS.VIEW Sets various view settings

OPTIONS.LISTS.ADD

This is the equivalent to clicking the Options command on the Tools menu and then clicking the Custom Lists tab in the Options diralog box. Used to add a new custom list.

Syntax

OPTIONS.LISTS.ADD(**string_array**)

OPTIONS.LISTS.ADD(**import_ref, by_row**)

OPTIONS.LISTS.ADD?(**import_ref, list_num**)

String_array is an array of strings or cell reference that contains the custom items in the list, a named cell reference, or an external reference containing the items of the custom list to add.

Import_ref is the reference to the cells that contain the members of the custom list. If A1:A12 contains the twelve signs of the Zodiac starting with Aquarius, then this function will add the contents of these twelve cells as a custom list.

By_row is a logical value that if TRUE, and if importing from cells, assumes that the list items are in sequential rows. If FALSE, assumes that the list items are in columns. If omitted, Microsoft Excel will try to determine the order of the custom lists according to the layout of the sheet.

List_num is a number specifying which list to activate. If omitted, then New List will be activated.

Remarks

- To replace an existing custom list, you must first delete it and then add the new list to the end.
- If the list already exists, then this function will do nothing. The list is not case sensitive, so "Scorpio" and "scorpio" are treated the same in custom lists.

Related Functions

OPTIONS.VIEW Sets various view settings

OPTIONS.LISTS.GET Returns contents of custom AutoFill lists

OPTIONS.LISTS.DELETE Deletes a custom list

OPTIONS.LISTS.DELETE

Equivalent to clicking the Options command on the Tools menu and then clicking the Delete button on the Custom Lists tab when a custom list is selected.

Syntax

OPTIONS.LISTS.DELETE(**list_num**)

List_num is the number of the custom list to delete. The first five lists (numbered zero through 4) cannot be deleted. If list_num doesn't exist, then FALSE is returned.

Related Functions

OPTIONS.LISTS.GET Returns contents of custom AutoFill lists

OPTIONS.LISTS.ADD Used to add a new custom list

OPTIONS.LISTS.GET

Returns contents of custom AutoFill lists as an array of text strings.

Syntax

OPTIONS.LISTS.GET(**list_num**)

OPTIONS.LISTS.GET(**string_array**)

List_num is a number specifying which list to return, as a horizontal string array. If list_num is zero, then FALSE is returned.

String_array is an array or cell reference to store the returned list.

Example

`OPTIONS.LIST.GET(3)` returns the twelve months of the year in the form {"Jan", "Feb", "Mar"}

Remarks

If list_num is zero or omitted, then FALSE is returned.

Related Functions

OPTIONS.LISTS.ADD Adds a new custom list

OPTIONS.LISTS.DELETE Deletes a custom list

OPTIONS.VIEW Sets various view settings

OPTIONS.TRANSITION

Equivalent to clicking the Options command on the Tools menu and then clicking the Transition tab in the Options dialog box. Sets options relating to compatibility with other spreadsheets.

Syntax

OPTIONS.TRANSITION(menu_key, menu_key_action, nav_keys, trans_eval, trans_entry)

OPTIONS.TRANSITION?(menu_key, menu_key_action, nav_keys, trans_eval, trans_entry)

Menu_key is text specifying which alternate menu key to use.

Menu_key_action is the number 1 or 2 specifying options for the alternate menu or Help key. In Microsoft Excel for the Macintosh, menu_key_action is ignored.

Menu_key_action	Alternate menu or Help key activates
1 or omitted	Microsoft Excel menus
2	Lotus 1-2-3 Help

Nav_keys is a logical value that corresponds to the Transition Navigation Keys check box, which if TRUE uses alternate navigation keys that correspond to the navigation keys for Lotus 1-2-3. In Microsoft Excel for the Macintosh, nav_keys is ignored.

Trans_eval is a logical value that corresponds to the Transition Formula Evaluation check box.

- If trans_eval is TRUE, Microsoft Excel uses a set of rules compatible with that of Lotus 1-2-3 when calculating formulas. Text is treated as 0. TRUE and FALSE are treated as 1 and 0. Certain characters in database criteria ranges are interpreted the same way Lotus 1-2-3 interprets them.
- If trans_eval is FALSE or omitted, Microsoft Excel calculates normally.

Trans_entry is a logical value that corresponds to the Transition Formula Entry check box.

- This argument is available only in Microsoft Excel for Windows.
- If trans_entry is TRUE, Microsoft Excel accepts formulas entered in Lotus 1-2-3 style.
- If trans_entry is FALSE or omitted, Microsoft Excel only accepts formulas entered in Microsoft Excel style.

Related Functions

OPTIONS.LISTS.DELETE	Deletes a custom list
OPTIONS.LISTS.GET	Returns contents of custom AutoFill lists
OPTIONS.VIEW	Sets various view settings

OPTIONS.VIEW

Equivalent to clicking the Options command on the Tools menu and then clicking the View tab in the Options dialog box. Sets various view settings.

Syntax

OPTIONS.VIEW(formula, status, notes, show_info, object_num, page_breaks, formulas, gridlines, color_num, headers, outline, zeros, hor_scroll, vert_scroll, sheet_tabs)

OPTIONS.VIEW?(formula, status, notes, show_info, object_num, page_breaks, formulas, gridlines, color_num, headers, outline, zeros, hor_scroll, vert_scroll, sheet_tabs)

Arguments correspond to check boxes and text boxes in the View tab on the Options dialog box. Arguments corresponding to check boxes are logical values. If an argument is TRUE, the check box is selected; if FALSE, the check box is cleared; if omitted, the current setting is not changed.

Formula   is a logical value corresponding to the Formula Bar check box. If TRUE, displays the formula bar. If FALSE, the formula bar is not displayed.

Status   is a logical value corresponding to the Status Bar check box. If TRUE, the status bar is displayed. If FALSE, the status bar is not displayed.

Notes   is a logical value corresponding to the Comment & Indicator check box. If TRUE, comments and comment indicators will be displayed. If FALSE, comments and indicators will not be displayed.

Show_info   is a logical value corresponding to the Info Window check box (only in Microsoft Excel 95 and earlier versions). If TRUE, displays the Info Window. The Info Window is not available in Microsoft Excel 97 or later.

Object_num   is a number from 1 to 3 corresponding to the display options in the Objects box.

Object_num	Corresponds to
1 or omitted	Show All
2	Show Placeholders

3	Hide

Page_breaks is a logical value corresponding to the Page Breaks check box. If TRUE, page breaks will appear. If FALSE, page breaks will not appear.

Formulas is a logical value corresponding to the Formulas check box. If TRUE, formulas will appear in the cells. If FALSE, formulas will not appear in the cells. The default is FALSE on worksheets and TRUE on macro sheets.

Gridlines is a logical value corresponding to the Gridlines check box. If TRUE, gridlines will be displayed. If FALSE, gridlines will not appear. The default is TRUE.

Color_num is a number from 0 to 56 corresponding to gridline color. Zero corresponds to automatic color and is the default value.

Headings is a logical value corresponding to the Row & Column Headers check box. If TRUE, row and column headers will be displayed. If FALSE, they will not be displayed. The default is TRUE.

Outline is a logical value corresponding to the Outline Symbols check box. If TRUE, outline symbols will appear. If FALSE, they will not appear. The default is TRUE.

Zeros is a logical value corresponding to the Zero Values check box. If TRUE, zero values will appear, If FALSE, zero values will not appear. The default is TRUE.

Hor_scroll is a logical value corresponding to the Horizontal Scroll Bar checkbox. If TRUE, the horizontal scroll bar will be displayed. If FALSE, it will not be displayed. The default is TRUE.

Vert_scroll is a logical value corresponding to the Vertical Scroll Bar checkbox. If TRUE, the vertical scroll bar will be displayed. If FALSE, it will not be displayed. The default is TRUE.

Sheet_tabs is a logical value corresponding to the Sheet Tabs check box. If TRUE, sheet tabs will be displayed. If FALSE, sheet tabs will not be displayed. The default is TRUE.

Related Functions

OPTIONS.LISTS.GET Returns contents of custom AutoFill lists

OPTIONS.LISTS.DELETE Deletes a custom list

OUTLINE

Creates an outline and defines settings for automatically creating outlines.

The first three arguments are logical values corresponding to check boxes in the Outline dialog box, which appears when you choose the Settings command from the Group and Outline submenu on the Data menu. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box. If an argument is omitted, the check box is left in its current state.

Syntax

OUTLINE(auto_styles, row_dir, col_dir, create_apply)

OUTLINE?(auto_styles, row_dir, col_dir, create_apply)

Auto_styles corresponds to the Automatic Styles check box.

Row_dir corresponds to the Summary Rows Below Detail check box.

Col_dir corresponds to the Summary Columns To Right Of Detail check box.

Create_apply is the number 1 or 2 and corresponds to the Create button and the Apply Styles button.

Create_apply	Result
1	Creates an outline with the current settings
2	Applies outlining styles to the selection based on outline levels
Omitted	Corresponds to choosing the OK button to set the other outline settings

Related Functions

DEMOTE Demotes the selection in an outline

PROMOTE Promotes the selection in an outline

OVERLAY

Equivalent to choosing the Overlay command from the Format menu in Microsoft Excel version 2.2 or earlier. This function is included only for macro compatibility. To format chart types in Microsoft Excel version 5.0 or later, use the FORMAT.CHART function.

Syntax

OVERLAY(**type_num**, stack, 100, vary, overlap, drop, hilo, overlap%, cluster, angle, series_num, auto)

Related Function

FORMAT.CHART Formats the chart according to the arguments you specify.

PAGE.SETUP

Equivalent to clicking the Page Setup command on the File menu. Use PAGE.SETUP to control the printed appearance of your sheets.

There are three syntax forms of PAGE.SETUP. Syntax 1 applies if a sheet or macro sheet is active; syntax 2 applies if a chart is active; syntax three applies to Visual Basic modules and the info Window.

Arguments correspond to check boxes and text boxes in the Page Setup dialog box. Arguments that correspond to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box. Arguments for margins are always in inches, regardless of your country setting.

Syntax 1

Worksheets and macro sheets

PAGE.SETUP(head, foot, left, right, top, bot, hdng, grid, h_cntr, v_cntr, orient, paper_size, scale, pg_num, pg_order, bw_cells, quality, head_margin, foot_margin, notes, draft)

PAGE.SETUP?(head, foot, left, right, top, bot, hdng, grid, h_cntr, v_cntr, orient, paper_size, scale, pg_num, pg_order, bw_cells, quality, head_margin, foot_margin, notes, draft)

Syntax 2

Charts

PAGE.SETUP(head, foot, left, right, top, bot, size, h_cntr, v_cntr, orient, paper_size, scale, pg_num, bw_chart, quality, head_margin, foot_margin, draft)

PAGE.SETUP?(head, foot, left, right, top, bot, size, h_cntr, v_cntr, orient, paper_size, scale, pg_num, bw_chart, quality, head_margin, foot_margin, draft)

Syntax 3

Visual Basic Modules and the Info Window

PAGE.SETUP(head, foot, left, right, top, bot, orient, paper_size, scale, quality, head_margin, foot_margin, pg_num)

PAGE.SETUP?(head, foot, left, right, top, bot, orient, paper_size, scale, quality, head_margin, foot_margin, pg_num)

Head specifies the text and formatting codes for the header for the current sheet . For information about formatting codes, see "Remarks" later in this topic.

Foot specifies the text and formatting codes for the workbook footer.

Left corresponds to the Left box and is a number specifying the left margin.

Right corresponds to the Right box and is a number specifying the right margin.

Top corresponds to the Top box and is a number specifying the top margin.

Bot corresponds to the Bottom box and is a number specifying the bottom margin.

Hdng corresponds to the Row & Column Headings check box. Hdng is available only in the sheet and macro sheet form of the function.

Grid corresponds to the Cell Gridlines check box. Grid is available only in the sheet and macro sheet form of the function.

H_cntr corresponds to the Center Horizontally check box in the Margins panel of the Page Setup dialog box.

V_cntr corresponds to the Center Vertically check box in the Margins panel of the Page Setup dialog box.

Orient determines the direction in which your workbook is printed.

Orient	Print format
1	Portrait
2	Landscape

Paper_size is a number from 1 to 26 that specifies the size of the paper.

Paper_size	**Paper type**
1	Letter
2	Letter (small)
3	Tabloid
4	Ledger
5	Legal
6	Statement
7	Executive
8	A3
9	A4
10	A4 (small)
11	A5
12	B4
13	B5
14	Folio
15	Quarto
16	10x14

17	11x17
18	Note
19	ENV9
20	ENV10
21	ENV11
22	ENV12
23	ENV14
24	C Sheet
25	D Sheet
26	E Sheet

Scale is a number representing the percentage to increase or decrease the size of the sheet. All scaling retains the aspect ratio of the original.

- To specify a percentage of reduction or enlargement, set scale to the percentage.
- For worksheets and macros, you can specify the number of pages that the printout should be scaled to fit. Set scale to a two-item horizontal array, with the first item equal to the width and the second item equal to the height. If no constraint is necessary in one direction, you can set the corresponding value to #N/A.
- Scale can also be a logical value. To fit the print area on a single page, set scale to TRUE.

Pg_num specifies the number of the first page. If zero, sets first page to zero. If "Auto" is used, then the page numbering is set to automatic. If omitted, PAGE.SETUP retains the existing pg_num.

Pg_order specifies whether pagination is left-to-right and then down, or top-to-bottom and then right.

Pg_order	Pagination
1	Top-to-bottom, then right
2	Left-to-right, then down

Bw_cells is a logical value that specifies whether to print cells and all graphic objects, such as text boxes and buttons, in color.

- If bw_cells is TRUE, Microsoft Excel prints cell text and borders in black and cell backgrounds in white.
- If bw_cells is FALSE , Microsoft Excel prints cell text, borders, and background patterns in color (or in gray scale).

Bw_chart is a logical value that specifies whether to print chart in color.

Size is a number corresponding to the options in the Chart Size box, and determines how you want the chart printed on the page within the margins. Size is available only in the chart form of the function.

Size	Size to print the chart
1	Screen size
2	Fit to page
3	Full page

Quality specifies the print quality in dots-per-inch. To specify both horizontal and vertical print quality, use an array of two values.

Head_margin is the placement, in inches, of the running head margin from the edge of the page.

Foot_margin is the placement, in inches, of the running foot margin from the edge of the page.

Draft corresponds to the Draft Quality checkbox in the Sheet tab and in the Chart tab of the Page Setup dialog box. If FALSE or omitted, graphics are printed with the sheet. If TRUE, no graphics are printed.

Notes specifies whether to print cell notes with the sheet. If TRUE, both the sheet and the cell notes are printed. If FALSE or omitted, just the sheet is printed.

Remarks

Microsoft Excel no longer requires you to enter formatting codes to format headers and footers, but the codes are still supported and recorded by the macro recorder. You can include these codes as part of the head and foot text strings to align portions of the header or footer to the left, right, or center; to include the page number, date, time, or workbook name; and to print the header or footer in bold or italic.

Formatting code	Result
&L	Left-aligns the characters that follow.
&C	Centers the characters that follow.
&R	Right-aligns the characters that follow.
&B	Turns bold printing on or off (now obsolete).
&I	Turns italic printing on or off.
&U	Turns single underlining printing on or off.
&S	Turns strikethrough printing on or off.
&O	Turns outline printing on or off (Macintosh only).
&H	Turns shadow printing on or off (Macintosh only).
&D	Prints the current date.
&T	Prints the current time.
&A	Prints the name of the sheet
&F	Prints the name of the workbook.
&P	Prints the page number.
&P+number	Prints the page number plus number.

&P-number	Prints the page number minus number.
&&	Prints a single ampersand.
& "fontname, fontstyle"	Prints the characters that follow in the specified font and style. Be sure to include a comma immediately following the fontname, and double quotation marks around fontname and fontstyle.
&nn	Prints the characters that follow in the specified font size. Use a two-digit number to specify a size in points.
&N	Prints the total number of pages in the workbook.
&E	Prints a double underline
&X	Prints the character as superscript
&Y	Prints the character as subscript

Related Functions

DISPLAY   Controls screen and Info Window display

GET.DOCUMENT   Returns information about a workbook

PRINT   Prints the active workbook

WORKSPACE   Changes workspace settings

PARSE

Distributes the contents of the current selection to fill several adjacent columns; the selection can be no more than one column wide. Use PARSE to reorganize data, especially data that you've read from files created by another application, such as a database.

Syntax

PARSE(parse_text, destination_ref)

PARSE?(parse_text, destination_ref)

Parse_text   is the parse line in the form of text. It is a copy of the first nonblank cell in the selected column, with square brackets indicating where to distribute (or parse) text. If parse_text is omitted, Microsoft Excel guesses where to place the brackets based on the spacing and formatting of data.

Destination_ref   is a reference to the upper-left corner of the range of cells where you want to place the parsed data. If destination_ref is omitted, it is assumed to be the current selection, so the parsed data will replace the original data.

Remarks

When you use the PARSE function, Microsoft Excel splits the first column into as many columns as you specify with parse_text and replaces any information in those columns.

PASTE

Equivalent to clicking the Paste command on the Edit menu. Pastes a selection or object that you copied or cut using the COPY or CUT function. Use PASTE when you want to paste all components of the selection. To paste only specific components of the selection, use PASTE.SPECIAL.

Syntax

PASTE(to_reference)

To_reference   is a reference to the cell or range of cells where you want to paste what you have copied. If to_reference is omitted, Microsoft Excel pastes to the current selection. If there is nothing to paste, the macro halts.

Related Functions

COPY	Copies and pastes data or objects
CUT	Cuts or moves data or objects
FORMULA	Enters values into a cell or range or onto a chart
INSERT	Inserts cells
PASTE.LINK	Pastes copied data and establishes a link to its source
PASTE.SPECIAL	Pastes specific components of copied data

PASTE.LINK

Equivalent to clicking the Paste Special command on the Edit menu, and then clicking the Paste Link button in the Paste Special dialog box. Pastes copied data or objects and establishes a link to the source of the data or object. The source can be either another Microsoft Excel workbook or another application. Use PASTE.LINK when you want Microsoft Excel to automatically update the paste area with any changes that occur in the source.

Syntax

PASTE.LINK( )

Note   To work properly, the application you are linking to must support dynamic data exchange (DDE) or object linking and embedding (OLE).

Related Functions

COPY	Copies and pastes data or objects
CUT	Cuts or moves data or objects
PASTE	Pastes cut or copied data

PASTE.SPECIAL   Pastes specific components of copied data

PASTE.PICTURE

Equivalent to clicking the Paste Picture command on the Edit menu while holding down the SHIFT key in Microsoft Excel version 4.0. Pastes a picture of the Clipboard contents onto the sheet. This picture is not linked, so changes to the source data will not be reflected in the picture. In Microsoft Excel for Windows version 5.0 or later, use INSERT.PICTURE to import pictures.

Syntax

PASTE.PICTURE( )

Related Functions

COPY.PICTURE   Creates a picture of the current selection for use in another program

INSERT.PICTURE   Inserts a picture from a file

PASTE.PICTURE.LINK   Pastes a linked picture of the currently copied area

PASTE.PICTURE.LINK

Equivalent to holding down the SHIFT key and clicking the Paste Picture Link command on the Edit menu in Microsoft Excel version 4.0 or to using the camera tool. Pastes a linked picture of the Clipboard contents. This picture is linked, so changes to the source data will be reflected in the picture.

Syntax

PASTE.PICTURE.LINK( )

Related Functions

COPY.PICTURE   Copies and pastes data or object

COPY.PICTURE   Creates a picture of the current selection for use in another program

CREATE.OBJECT   Creates an object

PASTE   Pastes cut or copied data

PASTE.PICTURE   Pastes a picture of the currently copied area

PASTE.SPECIAL

Equivalent to clicking the Paste Special command on the Edit menu. Pastes the specified components from the copy area into the current selection. The PASTE.SPECIAL function has four syntax forms.

Syntax 1   Pasting into a sheet or macro sheet

Syntax 2   Copying from a sheet and pasting into a chart.

Syntax 3   Copying and pasting between charts

Syntax 4   Pasting information from another application.

PASTE.SPECIAL SYNTAX 1

Equivalent to clicking the Paste Special command on the Edit menu. Pastes the specified components from the copy area into the current selection. The PASTE.SPECIAL function has four syntax forms. Use syntax 1 if you are pasting into a sheet or macro sheet.

Syntax

PASTE.SPECIAL(paste_num, operation_num, skip_blanks, transpose)

PASTE.SPECIAL?(paste_num, operation_num, skip_blanks, transpose)

Paste_num is a number from 1 to 6 specifying what to paste. Paste_num can also be quoted text of the object you want to paste.

Paste_num	Pastes
1	All
2	Formulas
3	Values
4	Formats
5	Comments
6	All except borders

Operation_num is a number from 1 to 5 specifying which operation to perform when pasting.

Operation_num	Action
1	None
2	Add
3	Subtract
4	Multiply

5	Divide

Skip_blanks is a logical value corresponding to the Skip Blanks check box in the Paste Special dialog box.

- If skip_blanks is TRUE, Microsoft Excel skips blanks in the copy area when pasting.
- If skip_blanks is FALSE, Microsoft Excel pastes normally.

Transpose is a logical value corresponding to the Transpose check box in the Paste Special dialog box.

- If transpose is TRUE, Microsoft Excel transposes rows and columns when pasting.
- If transpose is FALSE, Microsoft Excel pastes normally.

Related Functions

FORMULA	Enters values into a cell or range or onto a chart
PASTE	Pastes cut or copied data
PASTE.LINK	Pastes copied data and establishes a link to its source
Syntax 2	Copying from a sheet and pasting into a chart.
Syntax 3	Copying and pasting between charts
Syntax 4	Pasting information from another application.

PASTE.SPECIAL SYNTAX 2

Equivalent to clicking the Paste Special command on the Edit menu on the Chart menu bar. Pastes the specified components from the copy area into a chart. The PASTE.SPECIAL function has four syntax forms. Use syntax 2 if you have copied from a sheet and are pasting into a chart.

Syntax

PASTE.SPECIAL(rowcol, titles, categories, replace, series)

PASTE.SPECIAL?(rowcol, titles, categories, replace, series)

Rowcol is the number 1 or 2 and specifies whether the values corresponding to a particular data series are in rows or columns. Enter 1 for rows or 2 for columns.

Titles is a logical value corresponding to the Series Names In First Column check box (or First Row, depending on the value of rowcol) in the Paste Special dialog box.

- If series is TRUE, Microsoft Excel selects the check box and uses the contents of the cell in the first column of each row (or first row of each column) as the name of the data series in that row (or column).

- If series is FALSE, Microsoft Excel clears the check box and uses the contents of the cell in the first column of each row (or first row of each column) as the first data point of the data series.

Categories is a logical value corresponding to the Categories (X Labels) In First Row (or First Column, depending on the value of rowcol) check box in the Paste Special dialog box.

- If categories is TRUE, Microsoft Excel selects the check box and uses the contents of the first row (or column) of the selection as the categories for the chart.

- If categories is FALSE, Microsoft Excel clears the check box and uses the contents of the first row (or column) as the first data series in the chart.

Replace is a logical value corresponding to the Replace Existing Categories check box in the Paste Special dialog box.

- If replace is TRUE, Microsoft Excel selects the check box and applies categories while replacing existing categories with information from the copied cell range.

- If replace is FALSE, Microsoft Excel clears the check box and applies new categories without replacing any old ones.

Series is a number specifying how cells are added to a chart.

Series	Added as
1	New series
2	New point(s)

Related Functions

Syntax 1 Pasting into a sheet or macro sheet

Syntax 3 Copying and pasting between charts

Syntax 4 Pasting information from another application

PASTE.SPECIAL SYNTAX 3

Equivalent to clicking the Paste Special command on the Edit menu on the Chart menu bar. Pastes the specified components from the copy area into a chart. The PASTE.SPECIAL function has four syntax forms. Use syntax 3 if you have copied from a chart and are pasting into a chart.

Syntax

PASTE.SPECIAL(paste_num)

PASTE.SPECIAL?(paste_num)

Paste_num is a number from 1 to 3 specifying what to paste.

Paste_num	Pastes
1	All (formats and data series)
2	Formats only
3	Formulas (data series) only

Related Functions

Syntax 1 Pasting into a sheet or macro sheet

Syntax 2 Copying from a sheet and pasting into a chart

Syntax 4 Pasting information from another application

PASTE.SPECIAL SYNTAX 4

Equivalent to clicking the Paste Special command on the Edit menu when pasting data from another application into Microsoft Excel. Use syntax 4 when pasting information from another application.

Syntax

PASTE.SPECIAL(format_text, pastelink_logical, display_icon_logical, icon_file, icon_number, icon_label)

PASTE.SPECIAL?(format_text, pastelink_logical, display_icon_logical, icon_file, icon_number, icon_label)

Format_text is text specifying the type of data you want to paste from the Clipboard.

- The valid data types vary depending on the application from which the data was copied. For example, if you're copying data from Microsoft Word, some of the data types are "Microsoft Document Object", "Picture", and "Text".
- For more information about object classes, see your Microsoft Windows or Apple system software documentation.

Pastelink_logical is a logical value specifying whether to link the pasted data to its source application.

- If pastelink_logical is TRUE, Microsoft Excel updates the pasted information whenever it changes in the source application.

- If pastelink_logical is FALSE or omitted, the information is pasted without a link.

- If Microsoft Excel or the source application does not support linking for the specified format_text, then pastelink_logical is ignored.

Display_icon_logical is a logical value that specifies whether you want an application's icon to be displayed on the worksheet instead of the linked data. Equivalent to the Display as Icon check box on the Paste Special dialog box. If TRUE, the application's icon will be displayed. If FALSE or omitted, the application's icon will not be displayed.

Icon_file is the name of the file (with an .EXE or .DLL extension) that contains the icon. If display_icon_logical is FALSE, this argument is ignored.

Icon_number is the number associated with the icon and corresponds to the icon's relative position within the Icon Drop Down list box on the Change Icon Dialog box, which appears when you click the Change Icon button in the Paste Special dialog box. If display_icon_logical is FALSE, this argument is ignored.

Icon_label is the caption that you want to appear below the icon, and is equivalent to the Caption text box on the Change Icon dialog box, which appears when you click the Change Icon button in the Paste Special dialog box. If display_icon_logical is FALSE, this argument is ignored.

Related Functions

Syntax 1	Pasting into a sheet or macro sheet
Syntax 2	Copying from a sheet and pasting into a chart
Syntax 3	Copying and pasting between charts

PASTE.TOOL

Pastes a button face from the Clipboard to a specified position on a toolbar.

Syntax

PASTE.TOOL(**bar_id, position**)

Bar_id specifies the number or name of the toolbar into which you want to paste the button face. For detailed information about bar_id, see ADD.TOOL.

Position specifies the position within the toolbar of the button on which you want to paste the button face. Position starts with 1 at the left side (if horizontal) or at the top (if vertical).

Related Function

COPY.TOOL	Copies a button face

PATTERNS

Equivalent to clicking the Patterns tab in the Format Cells dialog box, which appears when you click the Cells command on the Format menu. Changes the appearance of the selected cells or objects or the selected chart item (you can select only one chart item at a time). The PATTERNS function has eight syntax forms: syntax 1 is for cells on a sheet or macro sheet. Syntax 2 is for lines or arrows on a worksheet, macro sheet, or chart. Syntax 3 is for objects on a sheet or macro sheet. Syntax 4 through syntax 8 are for chart items.

Syntax 1

Cells

PATTERNS(apattern, afore, aback, newui)

PATTERNS?(apattern, afore, aback, newui)

Syntax 2

Lines (arrows) on worksheets or charts

PATTERNS(lauto, lstyle, lcolor, lwt, hwidth, hlength, htype)

PATTERNS?(lauto, lstyle, lcolor, lwt, hwidth, hlength, htype)

Syntax 3

Text boxes, rectangles, ovals, arcs, and pictures on worksheets or macro sheets

PATTERNS(bauto, bstyle, bcolor, bwt, shadow, aauto, apattern, afore, aback, rounded, newui)

PATTERNS?(bauto, bstyle, bcolor, bwt, shadow, aauto, apattern, afore, aback, rounded, newui)

Syntax 4

Chart plot areas, bars, columns, pie slices, and text labels

PATTERNS(bauto, bstyle, bcolor, bwt, shadow, aauto, apattern, afore, aback, invert, apply, new_fill)

PATTERNS?(bauto, bstyle, bcolor, bwt, shadow, aauto, apattern, afore, aback, invert, apply, new_fill)

Syntax 5

Chart axes

PATTERNS(lauto, lstyle, lcolor, lwt, tmajor, tminor, tlabel)

PATTERNS?(lauto, lstyle, lcolor, lwt, tmajor, tminor, tlabel)

Syntax 6

Chart gridlines, hi-lo lines, drop lines, lines on a picture line chart, and picture charts of bar and column charts

PATTERNS(lauto, lstyle, lcolor, lwt, apply, smooth)

PATTERNS?(lauto, lstyle, lcolor, lwt, apply, smooth)

Syntax 7

Chart data lines

PATTERNS(lauto, lstyle, lcolor, lwt, mauto, mstyle, mfore, mback, apply, smooth)

PATTERNS?(lauto, lstyle, lcolor, lwt, mauto, mstyle, mfore, mback, apply, smooth)

Syntax 8

Picture chart markers

PATTERNS(type, picture_units, apply)

PATTERNS?(type, picture_units, apply)

The following argument descriptions are in alphabetic order. Arguments correspond to check boxes, list boxes, and options in the Patterns tab of the Format Cells dialog box for the selected item. The default for each argument reflects the setting in the dialog box.

Aauto is a number from 0 to 2 specifying area settings (that is, the object's "surface area").

If aauto is	Area settings are
0	Set by the user (custom)
1	Automatic (set by Microsoft Excel)
2	None

Aback is a number from 1 to 56 corresponding to the 56 area background colors in the Patterns tab of the Format Cells dialog box.

Afore is a number from 1 to 56 corresponding to the 56 area foreground colors in the Patterns tab of the Format Cells dialog box.

Apattern is a number corresponding to the area patterns in the Patterns tab of the Format Cells or Format Object dialog box. If an object is selected, apattern can be from 1 to 18; if a cell is selected, apattern can be from 0 to 18. If apattern is 0 and a cell is selected, Microsoft Excel applies no pattern.

Apply is a logical value corresponding to the Apply To All check box in Microsoft Excel version 4.0. This argument is for compatibility with previous versions of Microsoft Excel and applies only when a chart data point or a data series is selected.

- If apply is TRUE, Microsoft Excel applies any formatting changes to all items that are similar to the selected item on the chart.
- If apply is FALSE, Microsoft Excel applies formatting changes only to the selected item on the chart.

Bauto is a number from 0 to 2 specifying border settings.

If bauto is	Border settings are
0	Set by the user (custom)
1	Automatic (set by Microsoft Excel)
2	None

Bcolor is a number from 1 to 56 corresponding to the 56 border colors in the Border tab of the Format Object or Format (chart element) dialog box.

Bstyle is a number from 1 to 8 corresponding to the eight border styles in the Border tab of the Format Object or Format (chart element) dialog box.

Bwt is a number from 1 to 4 corresponding to the four border weights in the Border tab of the Format Object or Format (chart element) dialog box.

If bwt is	Border is
1	Hairline
2	Thin
3	Medium
4	Thick

Hlength is a number from 1 to 3 specifying the length of the arrowhead.

If hlength is	Arrowhead is
1	Short
2	Medium
3	Long

Htype is a number from 1 to 5 specifying the style of the arrowhead.

If htype is	Style of arrowhead is
1	No head
2	Open head
3	Closed head
4	Double open head
5	Double closed head

Hwidth is a number from 1 to 3 specifying the width of the arrowhead.

If hwidth is	Arrowhead is
1	Narrow
2	Medium
3	Wide

Invert is a logical value corresponding to the Invert If Negative check box in the Patterns tab of the Format Data Series dialog box. This argument applies only to data markers.

- If invert is TRUE, Microsoft Excel inverts the pattern in the selected item if it corresponds to a negative number.
- If invert is FALSE, Microsoft Excel removes the inverted pattern, if present, from the selected item corresponding to a negative value.

Lauto is a number from 0 to 2 specifying line settings.

If lauto is	Line settings are
0	Set by the user (custom)

1	Automatic (set by Microsoft Excel)
2	None

Lcolor is a number from 1 to 56 corresponding to the 16 line colors in the Patterns tab of the Format Object or Format (chart element) dialog box.

Lstyle is a number from 1 to 8 corresponding to the eight line styles in the Patterns tab of the Format Object or Format (chart element) dialog box.

Lwt is a number from 1 to 4 corresponding to the four line weights in the Patterns tab of the Format Object or Format (chart element) dialog box.

If lwt is	Line is
1	Hairline
2	Thin
3	Medium
4	Thick

Mauto is a number from 0 to 2 specifying marker settings.

If mauto is	Marker settings are
0	Set by the user
1	Automatic (set by Microsoft Excel)
2	None

Mback is a number from 1 to 56corresponding to the 56 marker background colors in the Patterns tab of the Format Data Series dialog box.

Mfore is a number from 1 to 56 corresponding to the 56 marker foreground colors in the Patterns tab of the Format Data Series dialog box.

Mstyle is a number from 1 to 9 corresponding to the nine marker styles in the Patterns tab of the Format Data Series dialog box.

Picture_units is the number of units you want each picture to represent in a scaled, stacked picture chart. This argument applies only to picture charts and only if type is 3.

Rounded is a logical value corresponding to the Round Corners check box and specifying whether to make the corners on text boxes and rectangles rounded. If rounded is TRUE, the corners are rounded; if FALSE, the corners are square. If the selection is an arc or an oval, rounded is ignored.

Newui is a logical value that specifies whether to use the foreground, background, and patterns of Microsoft Excel version 5.0 or later. If TRUE or omitted, the colors and patterns of Microsoft Excel version 5.0 or later will be used. If FALSE, the colors and patterns of Microsoft Excel version 4.0 will be used.

Newfill is a logical value that specifies whether to use the chart patterns of Microsoft Excel version 5.0 or later. If TRUE or omitted, the chart patterns of Microsoft Excel version 5.0 or later will be used. If FALSE, the chart patterns of Microsoft Excel version 4.0 will be used.

Shadow is a logical value corresponding to the Shadow check box. Shadow does not apply to area charts or bars in bar charts. If shadow is TRUE, Microsoft Excel adds a shadow to the selected item; if FALSE, Microsoft Excel removes the shadow, if one is present, from the selected item. If the selection is an arc, shadow is ignored.

Smooth is a logical value that applies smoothing to picture markers in line or xy (scatter) charts. The default is FALSE.

Tlabel is a number from 1 to 4 specifying the position of tick labels.

If tlabel is	Tick label position is
1	None
2	Low
3	High
4	Next to axis

Tmajor is a number from 1 to 4 specifying the type of major tick marks.

If tmajor is	Type of major tick marks is
1	None
2	Inside
3	Outside

4	Cross

Tminor is a number from 1 to 4 specifying the type of minor tick marks.

If tminor is	Type of minor tick marks is
1	None
2	Inside
3	Outside
4	Cross

Type is a number from 1 to 3 specifying the type of pictures to use in a picture chart.

If type is	Pictures should be
1	Stretched to reach a particular value
2	Stacked on top of each other to reach a particular value
3	Stacked on top of each other, but you specify the number of units each picture represents

Remarks

- You can select many graphic objects on a sheet or macro sheet and apply formatting to them at the same time, but you can select only one chart item at a time.
- If you select multiple objects and if one or more of the objects requires a different form of the PATTERNS function, then choose the syntax corresponding to the object with the most formatting attributes—that is, the syntax with the most arguments. If you specify an argument that does not apply to an item, the argument has no effect on that item.
- To apply formatting to similar items on a chart, use the apply argument described above.

Related Functions

FONT.PROPERTIES   Applies a font to the selection

FORMAT.TEXT   Formats a text box or a chart text item

PAUSE

Pauses a macro. Use the PAUSE function, instead of clicking the Pause button in the Single Step dialog box, as a debugging tool when you do not wish to step through a macro. You can also use PAUSE to enter and edit data, to work directly with Microsoft Excel commands, or to perform other actions that are not normally available when a macro is running.

Syntax

PAUSE(no_tool)

No_tool   is a logical value specifying whether to display the Resume Macro button when the macro is paused. If no_tool is TRUE, the toolbar is not displayed; if FALSE, the toolbar is displayed; if omitted, the toolbar is displayed unless you previously clicked the close box on the toolbar.

Remarks

- All commands and tools that are available when no macro is running are still available when a macro is paused.
- You can run other macros while a macro is paused, but you can pause only one macro at a time. If a macro is paused when you run a second macro containing a PAUSE function, Macro Resume resumes only the second macro; you cannot resume or return to the first macro automatically.
- PAUSE is ignored in custom worksheet functions, unless you manually run them by clicking the Run button in the Macro dialog box, which appears when you click the Macro command on the Tools menu. PAUSE is also ignored if it's placed in a formula for which the resume behavior would be unclear, such as:
- `IF(Cost<10, AND(PAUSE(),SUM(!$A$1:$A$4)))`
- If one macro runs a second macro that pauses, Microsoft Excel locks the calling cell in the first macro. If you try to edit this cell, Microsoft Excel displays an error message.
- To resume a paused macro, click the Resume Macro button on the toolbar or run a macro containing a RESUME function.
- If one macro runs a second macro that pauses and you need to halt only the paused macro, use RESUME(2) instead of HALT. HALT halts all macros and prevents resuming or returning to any macro. For more information, see RESUME.

Tip   Since the automatic Resume Macro button can be customized, you can create a custom toolbar that will appear whenever a macro pauses.

Example

The following macro formula checks to see if a variable named TestValue is greater than 9. If it is, the macro pauses; otherwise, the macro continues normally.

```
IF(TestValue>9, PAUSE())
```

Related Functions

HALT Stops all macros from running

RESUME Resumes a paused macro

STEP Turns on macro single-stepping

PIVOT.ADD.DATA

Adds a field to a PivotTable report.

Syntax

PIVOT.ADD.DATA(name, pivot_field_name, new_name, position, function, calculation, base_field, base_item, format_text)

Name is the name of the PivotTable report to which the user wants to add as a data field. If name is omitted, Microsoft Excel will use the PivotTable report containing the active cell.

Pivot_field_name is the name of a field which the user would like to add to the PivotTable report as data or as text.

New_name is the name you would like to give to the new field once it is added to your PivotTable report. If this argument is omitted, Microsoft Excel will pick a default name for you. This function returns new_name or the name Microsoft Excel gives the field.

Position is the position within all the Data fields you would like to place the new data field. If position is omitted, the field will be added as the last data field.

Function is a number from 2 to 2048 specifying how the new field is to be calculated. To compute the value to place in this column use one value from the following table. If function is omitted, SUM will be used. If the field is a numeric field or text field, COUNTA will be used.

Value	Function
2	SUM
4	COUNTA
8	COUNT
16	AVERAGE
32	MAX
64	MIN

128	PRODUCT
256	STDEV
512	STDEVP
1024	VAR
2048	VARP

Calculation is a number between 1 and 9 representing which custom calculation you would like to apply to this data field. This corresponds to the Show Data As drop-down box on the PivotTable Field dialog box. If this argument is omitted, no special calculation will be applied to the data field.

Value	Calculation
1	Normal
2	Difference From
3	% Of Item
4	% Difference From
5	Running Total In
6	% of Row
7	% of Column
8	% of Total
9	Index

Base_Field is the field on which you want to base the calculation.

Base_Item is the item within base_field on which you want to base the calculation.

Format_text is the type of number format you want to apply to the PivotTable data. Corresponds to the number button in the PivotTable Field dialog box, which appears when you click the PivotTable Field command on the Data menu when the selection is in a data field.

Remarks

- If name is not a valid PivotTable name, then the #VALUE! error value is returned.
- If field_name is not a valid field for the current PivotTable report then the #VALUE! error value is returned.

Related Functions

PIVOT.ADD.FIELDS	Adds fields to a PivotTable report
PIVOT.FIELD	Pivots fields within a PivotTable report
PIVOT.FIELD.GROUP	Creates groups within a PivotTable report
PIVOT.FIELD.PROPERTIES	Changes the properties of a field inside a PivotTable report
PIVOT.FIELD.UNGROUP	Ungroups all selected groups within a PivotTable report
PIVOT.ITEM	Moves an item within a PivotTable report
PIVOT.ITEM.PROPERTIES	Changes the properties of an item within a header field
PIVOT.REFRESH	Refreshes a PivotTable report
PIVOT.SHOW.PAGES	Creates new sheets in the workbook containing the active cell
PIVOT.TABLE.WIZARD	Creates an empty PivotTable report

PIVOT.ADD.FIELDS

Add fields onto a PivotTable report.

Syntax

PIVOT.ADD.FIELDS(name, row_array, column_array, page_array, add_to_table)

Name is the name of the PivotTable report to which the user wants to add fields. If name is omitted, Microsoft Excel will use the PivotTable report containing the active cell.

Row_array is an array of text constants consisting of the names of the fields which the user would like to add to the PivotTable report as row fields.

Column_array is an array of text constants consisting of the names of the fields which the user would like to add to the PivotTable report as column fields.

Page_array is an array of text constants consisting of the names of the fields which the user would like to add to the PivotTable report as page fields.

Add_to_table is a logical value which if TRUE adds the fields specified by row_array, column_array and page_array to the existing fields on the PivotTable report. If add_to_table is FALSE, Microsoft Excel will replace the fields already along the rows, columns and pages with the fields specified by row_array, column_array and page_array.

Remarks

If name is not a valid PivotTable name, then the #VALUE! error value is returned.

Related Functions

PIVOT.ADD.DATA	Adds a field to a PivotTable report as a data field
PIVOT.FIELD	Pivots fields within a PivotTable report
PIVOT.FIELD.GROUP	Creates groups within a PivotTable report
PIVOT.FIELD.PROPERTIES	Changes the properties of a field inside a PivotTable report
PIVOT.FIELD.UNGROUP	Ungroups all selected groups within a PivotTable report
PIVOT.ITEM	Moves an item within a PivotTable report
PIVOT.ITEM.PROPERTIES	Changes the properties of an item within a header field
PIVOT.REFRESH	Refreshes a PivotTable report
PIVOT.SHOW.PAGES	Creates new sheets in the workbook containing the active cell
PIVOT.TABLE.WIZARD	Creates an empty PivotTable report

PIVOT.FIELD

Pivots a field within a PivotTable report.

Syntax

PIVOT.FIELD(name, pivot_field_name, orientation, position)

Name is the name of the PivotTable report in which the user wants to pivot fields. If name is omitted, Microsoft Excel will use the PivotTable report containing the active cell.

Pivot_field_name is the name of the field which the user wishes to pivot to another part of the PivotTable report. This argument is given as a text constant or a reference to a text constant. If field_name is omitted, Microsoft Excel uses the field containing the active cell.

Orientation is an integer representing the destination of the field which is being pivoted. If this argument is omitted, then the orientation remains unchanged. The integers refer to orientations as follows:

Value	Orientation
0	Hidden
1	Row
2	Column
3	Page
4	Data

Position is an integer representing where in the orientation the fields will be positioned. Position 1 is the leftmost header position in the row header and the topmost position in the column header. This argument is ignored if orientation is set to 0. If the position argument is omitted, it will default to the last position in the field.

Remarks

- The function returns TRUE if successful.
- If name is not a valid PivotTable name then the #VALUE! error value is returned.
- If pivot_field_name is not a text constant or contains text which is not a valid field name for the PivotTable report then the #VALUE! error value is returned.
- If destination is not an integer between 0 and 4, then the #VALUE! error value is returned.

Related Functions

PIVOT.ADD.DATA	Adds a field to a PivotTable report as a data field
PIVOT.ADD.FIELDS	Adds fields to a PivotTable report
PIVOT.FIELD.GROUP	Creates groups within a PivotTable report
PIVOT.FIELD.PROPERTIES	Changes the properties of a field inside a PivotTable report
PIVOT.FIELD.UNGROUP	Ungroups all selected groups within a PivotTable report
PIVOT.ITEM	Moves an item within a PivotTable report
PIVOT.ITEM.PROPERTIES	Changes the properties of an item within a header field
PIVOT.REFRESH	Refreshes a PivotTable report
PIVOT.SHOW.PAGES	Creates new sheets in the workbook containing the active cell
PIVOT.TABLE.WIZARD	Creates an empty PivotTable report

PIVOT.FIELD.GROUP

Creates groups within a PivotTable report.

Syntax

PIVOT.FIELD.GROUP(start, end, by, periods)

PIVOT.FIELD.GROUP?(start, end, by, periods)

Start is the beginning date of the range to be grouped. If start is TRUE or omitted, it is assumed to be the first value in the field.

End is the ending date of the range to be grouped. If end is TRUE or omitted it is assumed to be the last value in the field.

By is the size of the groups to be created. If by is omitted, Microsoft Excel uses a default group size. If grouping a date field and if periods is not 8(days) then by is ignored.

Periods is a number between 1 and 127. It is calculated by summing the values in the following table corresponding to the periods into which you want to group your dates. This argument is ignored if the field is not a date field. This argument takes precedence over By if they are both specified for a date field.

Value	Periods
1	Seconds
2	Minutes
4	Hours
8	Days
16	Months
32	Quarters
64	Years

Remarks

- This function returns TRUE if the grouping is successful. The #N/A error value is returned if the grouping failed.
- If no arguments are specified and multiple items within the header field are selected then this function groups those selected items.
- If no arguments are specified and a single item within the header field is selected then the #VALUE! error value is returned.

Related Functions

PIVOT.ADD.DATA	Adds a field to a PivotTable report as a data field
PIVOT.ADD.FIELDS	Adds fields to a PivotTable report
PIVOT.FIELD	Pivots fields within a PivotTable report
PIVOT.FIELD.PROPERTIES	Changes the properties of a field inside a PivotTable report
PIVOT.FIELD.UNGROUP	Ungroups all selected groups within a PivotTable report
PIVOT.ITEM	Moves an item within a PivotTable report
PIVOT.ITEM.PROPERTIES	Changes the properties of an item within a header field

PIVOT.REFRESH Refreshes a PivotTable report

PIVOT.SHOW.PAGES Creates new sheets in the workbook containing the active cell

PIVOT.TABLE.WIZARD Creates an empty PivotTable report

PIVOT.FIELD.PROPERTIES

Changes the properties of a field inside a PivotTable report.

Syntax

PIVOT.FIELD.PROPERTIES(name, pivot_field_name, new_name, orientation, function, formats)

PIVOT.FIELD.PROPERTIES?(name, pivot_field_name, new_name, orientation, function, formats)

Name is the name of the PivotTable report containing the field which the user wants to edit. If name is omitted, Microsoft Excel will use the PivotTable report containing the active cell.

Pivot_field_name is the name of a field in the PivotTable report which the user would like to edit, as text. If it is omitted, Microsoft Excel uses the field containing the active cell.

New_name is the name which you would like to rename the current field. If it is omitted, the name of the current field will not change.

Orientation is a number between 0 and 4 specifying which area will show the field containing the active cell. If zero, then the field is deleted and all other arguments to this function are ignored. If this argument is omitted, the orientation of the field will not change.

Value	Orientation
0	Delete
1	Display as row
2	Display as column
3	Display as page
4	Display as data

Function is a number between 0 and 4094 specifying which calculation or subtotals to apply to the field. If you will be showing the field in the header (orientation 1, 2, or 3), add up the values from the table corresponding to the subtotals you would like to show. If you will be showing the field as a data field (orientation 4), use one value from the table. If an entry in this column is left blank, Microsoft Excel will not change the calculation or subtotal which are currently attached to the field.

Value	Function
0	NO SUBTOTALS
1	AUTOMATIC
2	SUM
4	COUNTA
8	COUNT
16	AVERAGE
32	MAX
64	MIN
128	PRODUCT
256	STDEV
512	STDEVP
1024	VAR
2048	VARP

Formats is either a one- or a two- dimensional array, depending on whether the field is a header field or a data field.

- If the active field is a header field (orientation argument is 1, 2 or 3) then this is a two-dimensional array. Each row of the array should consist of two entries. The first is a text string corresponding to the item whose property is being changed. The second element specifies whether the item will be hidden. If this argument is TRUE, the item will be hidden and therefore will not be displayed in the PivotTable report. If the argument is FALSE, then the item will be displayed in the PivotTable report.

- If the active field is a data field, then the array is a one-dimensional array with four elements. The first element is a number between 1 and 9 specifying which calculation you wish to apply to the current data field. This corresponds to the Show Data As drop-down box on the PivotTable Field dialog box.

Value	Format
1	Normal
2	Difference From
3	%Of Item
4	%Difference From
5	Running Total In
6	%Of Row
7	%Of Column
8	%Of Subtotal
9	Index

- The second element contains a text string representing the field to which your data field is related. This argument is not necessary for the Normal calculation. If omitted, Microsoft Excel will use the first field that would appear in the Base Field list box.

- The third element must contain a text string representing an item in the base field on which to base your calculation. Note that this argument is not necessary for calculations like Running Total In which relies only on a Base Field. If omitted, Microsoft Excel will use the first item that would appear in the Base Item list box.

- The fourth element is a text string representing the number format you wish to apply to the data field.

Remarks

- If pivot_field_name is not a valid field name for the PivotTable report then the #VALUE! error value is returned.
- If name is not a valid PivotTable name, then the #VALUE! error value is returned.
- If the orientation and function arguments do not contain numbers or if these arguments contain numbers which are out of range then the #VALUE! error value is returned.

Related Functions

PIVOT.ADD.DATA	Adds a field to a PivotTable report as a data field
PIVOT.ADD.FIELDS	Adds fields to a PivotTable report
PIVOT.FIELD	Pivots fields within a PivotTable report
PIVOT.FIELD.GROUP	Creates groups within a PivotTable report
PIVOT.FIELD.UNGROUP	Ungroups all selected groups within a PivotTable report
PIVOT.ITEM	Moves an item within a PivotTable report
PIVOT.ITEM.PROPERTIES	Changes the properties of an item within a header field
PIVOT.REFRESH	Refreshes a PivotTable report
PIVOT.SHOW.PAGES	Creates new sheets in the workbook containing the active cell
PIVOT.TABLE.WIZARD	Creates an empty PivotTable report

PIVOT.FIELD.UNGROUP

Ungroups all selected groups within a PivotTable report.

Syntax

PIVOT.FIELD.UNGROUP( )

Remark

If the active cell is on a field header, then all the groups in that field are ungrouped and the field will be removed from the PivotTable report. Similarly, if the last group in a Parent field is ungrouped, the entire field will be removed from the PivotTable report.

Related Functions

PIVOT.ADD.DATA	Adds a field to a PivotTable report as a data field
PIVOT.ADD.FIELDS	Adds fields to a PivotTable report
PIVOT.FIELD	Pivots fields within a PivotTable report
PIVOT.FIELD.GROUP	Creates groups within a PivotTable report
PIVOT.FIELD.PROPERTIES	Changes the properties of a field inside a PivotTable report
PIVOT.ITEM	Moves an item within a PivotTable report
PIVOT.ITEM.PROPERTIES	Changes the properties of an item within a header field

PIVOT.REFRESH   Refreshes a PivotTable report

PIVOT.SHOW.PAGES   Creates new sheets in the workbook containing the active cell

PIVOT.TABLE.WIZARD   Creates an empty PivotTable report

PIVOT.ITEM

Moves an item within a PivotTable report.

Syntax

PIVOT.ITEM(name, pivot_field_name, pivot_item_name,position)

Name   is the name of the PivotTable report within which an item will be repositioned. If name is omitted, Microsoft Excel will use the PivotTable report containing the active cell.

Pivot_field_name   is the name of the field within which an item will be repositioned, given as a text string. If pivot_field_name is omitted, Microsoft Excel will use the field containing the active cell. If the active cell is not within a field, then this argument is required.

Pivot_item_name   is the name of the item to be repositioned in its field (given as a text constant). If it is omitted, Microsoft Excel uses the item containing the active cell. If the active cell is not contained within an item, then this argument is required.

Position   is a number representing where in the field the items will be moved. Position 1 is the topmost position within the row field and the leftmost position within the column field and the highest position within the page field. If the position argument is omitted, it will default to the last position in the field.

Remarks

- If an item is set to be visible, but its display is suppressed because there is no data, this item still occupies a valid position.
- If name is not a valid PivotTable name then the #VALUE! error value is returned.
- If pivot_field_name is not a text string, or if pivot_field_name is not a text string within a valid field name, then #VALUE! is returned.
- If pivot_item_name is an item which is not currently showing in the PivotTable report because it does not exist in the field pivot_field_name, the #VALUE! error value is returned.

Related Functions

PIVOT.ADD.DATA   Adds a field to a PivotTable report as a data field

PIVOT.ADD.FIELDS   Adds fields to a PivotTable report

PIVOT.FIELD   Pivots fields within a PivotTable report

PIVOT.FIELD.GROUP   Creates groups within a PivotTable report

PIVOT.FIELD.PROPERTIES   Changes the properties of a field inside a PivotTable report

PIVOT.FIELD.UNGROUP   Ungroups all selected groups within a PivotTable report

PIVOT.ITEM.PROPERTIES   Changes the properties of an item within a header field

PIVOT.REFRESH   Refreshes a PivotTable report

PIVOT.SHOW.PAGES   Creates new sheets in the workbook containing the active cell

PIVOT.TABLE.WIZARD   Creates an empty PivotTable report

PIVOT.ITEM.PROPERTIES

Changes the properties of an item within a header field.

Syntax

PIVOT.ITEM.PROPERTIES(name, pivot_field_name, pivot_item_name, new_name, position, show, active_page)

Name   is the name of the PivotTable report containing the item which the user wants to edit.

Pivot_field_name   is the name of a field in the PivotTable report containing the item which the user would like to edit. If it is omitted, Microsoft Excel uses the field containing the active cell.

Pivot_item_name   is the name of the item which the user would like to edit. If it is omitted, Microsoft Excel uses the item containing the active cell.

New_name   is the name which you would like to rename the current item. If it is omitted, then the name of the current item will not change.

Position   is a number representing where in the field the item will appear. Position 1 is the topmost position within the row field, the leftmost position within the column field, and the highest position within the page field. If the position argument is omitted, it will default to the last position in the field.

Show   is a logical value which if TRUE causes the item to appear in the PivotTable report. If FALSE, the item will be hidden.

Active_page   is a logical value specifying whether item_name will become the active item in the page field. If TRUE then item_name will become the active item in the page field. If FALSE or omitted, the active item in the page field does not change. Applies to only page fields.

Remarks

- If name is omitted, Microsoft Excel will use the PivotTable report containing the active cell.
- If pivot_field_name is not a header field, then this function will return the #VALUE! error value.

Related Functions

PIVOT.ADD.DATA   Adds a field to a PivotTable report as a data field

PIVOT.ADD.FIELDS   Adds fields to a PivotTable report

PIVOT.FIELD   Pivots fields within a PivotTable report

PIVOT.FIELD.GROUP   Creates groups within a PivotTable report

PIVOT.FIELD.PROPERTIES   Changes the properties of a field inside a PivotTable report

PIVOT.FIELD.UNGROUP   Ungroups all selected groups within a PivotTable report

PIVOT.ITEM   Moves an item within a PivotTable report

PIVOT.REFRESH   Refreshes a PivotTable report

PIVOT.SHOW.PAGES   Creates new sheets in the workbook containing the active cell

PIVOT.TABLE.WIZARD   Creates an empty PivotTable report

PIVOT.REFRESH

Refreshes a PivotTable report.

Syntax

PIVOT.REFRESH(name)

Name   is the name of the PivotTable report the user would like to refresh with current data. If name is omitted, Microsoft Excel will use the PivotTable report containing the active cell.

Remarks

- If the function is successful, it returns TRUE; otherwise, it returns the #VALUE! error value.
- If name is not a valid PivotTable name, then the #VALUE! error value is returned.

Related Functions

PIVOT.ADD.DATA   Adds a field to a PivotTable report as a data field

PIVOT.ADD.FIELDS   Adds fields to a PivotTable report

PIVOT.FIELD   Pivots fields within a PivotTable report

PIVOT.FIELD.GROUP   Creates groups within a PivotTable report

PIVOT.FIELD.PROPERTIES   Changes the properties of a field inside a PivotTable report

PIVOT.FIELD.UNGROUP   Ungroups all selected groups within a PivotTable report

PIVOT.ITEM   Moves an item within a PivotTable report

PIVOT.ITEM.PROPERTIES   Changes the properties of an item within a header field

PIVOT.SHOW.PAGES   Creates new sheets in the workbook containing the active cell

PIVOT.TABLE.WIZARD   Creates an empty PivotTable report

PIVOT.SHOW.PAGES

Creates new sheets in the workbook containing the active cell. The function will iterate through each item in page_field and create a new PivotTable report on a new sheet with the page field set to that particular item.

Syntax

PIVOT.SHOW.PAGES(name, page_field)

Name   is the name of the target PivotTable report. If name is omitted, Microsoft Excel will use the PivotTable report containing the active cell.

Page_field   is the name of a page field in the PivotTable report specified by the name argument.

Remarks

- If the function is successful, it returns TRUE; otherwise, it returns the #VALUE! error value.

- If name is not a valid PivotTable name then the #VALUE! error value is returned.

Related Functions

PIVOT.ADD.DATA	Adds a field to a PivotTable report as a data field
PIVOT.ADD.FIELDS	Adds fields to a PivotTable report
PIVOT.FIELD	Pivots fields within a PivotTable report
PIVOT.FIELD.GROUP	Creates groups within a PivotTable report
PIVOT.FIELD.PROPERTIES	Changes the properties of a field inside a PivotTable report
PIVOT.FIELD.UNGROUP	Ungroups all selected groups within a PivotTable report
PIVOT.ITEM	Moves an item within a PivotTable report
PIVOT.ITEM.PROPERTIES	Changes the properties of an item within a header field
PIVOT.REFRESH	Refreshes a PivotTable report
PIVOT.TABLE.WIZARD	Creates an empty PivotTable report

PIVOT.TABLE.WIZARD

Creates an empty PivotTable report.

Syntax

PIVOT.TABLE.WIZARD(type, source, destination, name, row_grand, col_grand, save_data, apply_auto_format, autopage)

PIVOT.TABLE.WIZARD?(type, source, destination, name, row_grand, col_grand, save_data, apply_auto_format, autopage)

Type   is a number specifying the type of source data used to create the PivotTable report.

Value	Type of source data
1	Microsoft Excel list or database
2	External data source
3	Multiple consolidation ranges
4	Another PivotTable report

Source   can be one of four things. If type is 1, then source is a cell reference or name to the range to be used as the PivotTable source. If type is 2, then source is a one-dimensional array describing the external database to be used as the PivotTable source. If type is 3, then source is a multi-dimensional array listing the cell ranges and associated page field items describing the consolidation PivotTable source. If type is 4, then source is the name of another PivotTable report with which to share its source.

Destination   is a cell reference or name. The upper-left corner of this range will act as the upper-left corner of the PivotTable report which will be created. If destination is omitted, Microsoft Excel will create the PivotTable report on a new sheet.

Name   is the name of the PivotTable report to be created given as a text. If name is omitted, Microsoft Excel uses a default name.

Row_grand   is a logical value which if TRUE displays a Grand Total for each row on the PivotTable report. If FALSE, a Grand Total for each row is not displayed.

Col_grand   is a logical value which if TRUE displays a Grand Total for each column. If FALSE, a Grand Total for each column is not displayed.

Save_data   is a logical value which if TRUE causes the data for the PivotTable report to be saved along with the PivotTable definition. If FALSE, the data is not saved along with the PivotTable definition.

Apply_auto_format   is a logical value which if TRUE causes autoformatting upon pivotting or refreshing. If FALSE, the PivotTable report will not be formatted automatically upon pivoting or refreshing.

Autopage   Applies only to type 3. This argument is a logical value which if TRUE or omitted causes Microsoft Excel to create a page field automatically. If FALSE, the page field must be created manually.

Remarks

- The function will return TRUE if successful; otherwise, returns the #VALUE! error value.
- If destination is not a valid Microsoft Excel reference, then #VALUE! error value is returned.
- If name is not a valid PivotTable name, then the #VALUE! error value is returned.

Related Functions

PIVOT.ADD.DATA   Adds a field to a PivotTable report as a data field

PIVOT.ADD.FIELDS   Adds fields to a PivotTable report

PIVOT.FIELD   Pivots fields within a PivotTable report

PIVOT.FIELD.GROUP   Creates groups within a PivotTable report

PIVOT.FIELD.PROPERTIES   Changes the properties of a field inside a PivotTable report

PIVOT.FIELD.UNGROUP   Ungroups all selected groups within a PivotTable report

PIVOT.ITEM   Moves an item within a PivotTable report

PIVOT.ITEM.PROPERTIES   Changes the properties of an item within a header field

PIVOT.REFRESH   Refreshes a PivotTable report

PIVOT.SHOW.PAGES   Creates new sheets in the workbook containing the active cell

PLACEMENT

Equivalent to choosing the Object Placement command from the Format menu in Microsoft Excel version 3.0. Determines how the selected object or objects are attached to the cells beneath them. This function is included only for macro compatibility and will be converted to OBJECT.PROPERTIES when you load older macro sheets. For more information, see OBJECT.PROPERTIES.

Syntax

PLACEMENT(placement_type)

PLACEMENT?(placement_type)

Related Functions

OBJECT.PROPERTIES   Determines an object's relationship to underlying cells

POKE

Sends data to another application. Use POKE to send data to documents in other applications you are communicating with through dynamic data exchange (DDE).

Syntax

POKE(**channel_num, item_text**, **data_ref**)

Important   Microsoft Excel for the Macintosh requires system software version 7.0 or later for this function.

Channel_num   is the channel number returned by a previously run INITIATE function.

Item_text   is text that identifies the item you want to send data to in the application you are accessing through channel_num. The form of item_text depends on the application connected to channel_num.

Data_ref   is a reference to the workbook containing the data to send.

If POKE is not successful, it returns the following values.

Value returned	Meaning
#VALUE!	Channel_num is not a valid channel number.
#DIV/0!	The application you are accessing does not respond after a certain length of time, and you press ESC to cancel.
#REF!	POKE is refused.

Examples

In Microsoft Excel for Windows, the following macro inserts the text from cell C3 into the Microsoft Word for Windows document SALES.DOC at the start of the document.

```
=POKE(SendChanl, "StartOfDoc", C3)
```

In Microsoft Excel for the Macintosh, the following macro inserts the text from cell C3 into the Microsoft Word document named Report.

```
=POKE(SendChanl, "TopicName", C3)
```

Related Functions

INITIATE	Opens a channel to another application
REQUEST	Returns data from another application
TERMINATE	Closes a channel to another application

PRECISION

Equivalent to selecting or clearing the Precision As Displayed check box in the Calculation tab of the Options dialog box, which appears when you click the Options command on the Tools menu. Controls how values are stored in cells. Use PRECISION when the results of formulas do not seem to match the values used to calculate the formulas.

Syntax

PRECISION(logical)

Logical is a logical value corresponding to the Precision As Displayed check box in the Calculation tab.

- If logical is TRUE, Microsoft Excel stores future entries at full precision (15 digits).
- If logical is FALSE or omitted, Microsoft Excel stores values exactly as they are displayed.

Caution The PRECISION function may permanently alter your data. PRECISION(FALSE) causes Microsoft Excel to change values on your worksheet or macro sheet to match displayed values. PRECISION(TRUE) causes Microsoft Excel to store future values at full precision, but it does not restore previously entered numbers to their original values.

Remarks

- Precision As Displayed does not affect numbers in General format. Numbers in General format are always calculated to full precision.
- Microsoft Excel calculates slightly faster when using full precision because with Precision As Displayed selected, Microsoft Excel has to round off numbers as it calculates.

Related Functions

FORMAT.NUMBER	Applies a number format to the selection
WORKSPACE	Changes workspace settings

PREFERRED

Equivalent to clicking the Preferred command on the Gallery menu in Microsoft Excel version 4.0. Changes the format of the active chart to the format currently defined by the Set As Default Chart option in the Standard Types tab of the Chart Type dialog box or the SET.PREFERRED macro function.

Syntax

PREFERRED( )

Related Function

SET.PREFERRED   Changes the default chart format

PRESS.TOOL

Formats a button so that it appears either normal or depressed into the screen.

Syntax

PRESS.TOOL(**bar_id, position**, down)

Bar_id   specifies the number or name of the toolbar in which you want to change the button appearance. For detailed information about bar_id, see ADD.TOOL.

Position   specifies the position of the button within the toolbar. Position starts with 1 at the left side (if horizontal) or at the top (if vertical).

Down   is a logical value specifying the appearance of the button. If down is TRUE, the button appears depressed into the screen; if FALSE or omitted, it appears normal (up).

Remarks

This function applies only to custom buttons to which macros have already been assigned. An error occurs if you try to process any other type of button.

Example

The following macro formula sets the third button image on Toolbar4 to normal (up).

```
PRESS.TOOL("Toolbar4", 3, FALSE)
```

Related Functions

ADD.TOOL   Adds one or more buttons to a toolbar

DELETE.TOOL   Deletes a button from a toolbar

PRINT

Equivalent to clicking the Print command on the File menu. Prints the active workbook.

Arguments correspond to options, check boxes, and edit boxes in the Print dialog box. Arguments corresponding to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box.

Syntax

PRINT(range_num, from, to, copies, draft, preview, print_what, color, feed, quality, y_resolution, selection)

PRINT?(range_num, from, to, copies, draft, preview, print_what, color, feed, quality, y_resolution, selection)

Range_num is a number specifying which pages to print.

Range_num	Prints the following pages
1	All the pages
2	Prints a specified range. If range_num is 2, then from and to are required arguments.

From specifies the first page to print. This argument is ignored unless range_num equals 2.

To specifies the last page to print. This argument is ignored unless range_num equals 2.

Copies specifies the number of copies to print. If omitted, the default is 1.

Draft This argument overrides the draft argument from the PAGE.SETUP function. If omitted, the Draft Setting from the Page.Setup function is used.

Preview is a logical value corresponding to the Print Preview button in the Print dialog box. If TRUE, the print preview window will be displayed. If FALSE, the window will not be displayed

Print_what is a number from 1 to 3 that specifies what parts of the sheet or macro sheet to print. If a chart is active, print_what is ignored. This argument will override the setting in the Page Setup dialog box. If omitted, the note argument in the Page.Setup function is used to determine whether to print notes or not.

Print_what	Prints
1	Sheet only
2	Notes only
3	Sheet and then notes

Color corresponds to the Print Using Color check box. Color is available only in Microsoft Excel for the Macintosh. If omitted, the setting is not changed.

Feed is a number specifying the type of paper feed. Feed is available only in Microsoft Excel for the Macintosh.

Feed	Type of paper feed
1 or omitted	Continuous (paper cassette)
2	Cut sheet or manual (manual feed)

Quality Specifies the DPI output quality you want. If omitted, the corresponding settings in the Page Setup dialog box will be used. If included, this argument overrides the quality argument in the PAGE SETUP dialog box.

Y_resolution corresponds to the Print Quality box in the Page Setup dialog box if you have specified a printer where the horizontal and vertical resolution are not equal, such as a dot-matrix printer. If omitted, the corresponding settings in the Page Setup dialog box will be used. If included, this argument overrides the print quality setting in the Page Setup dialog box.

Selection specifies what portion of the sheet to print.

Selection	Portion printed
1	Prints the current selection from all selected sheets. For example, if A1:F40 is selected on the active sheet, A1:F40 will be printed from each of the selected sheets.
2	Prints the print area or entire sheet from all selected sheets.
3	Prints print area or entire sheet from all sheets in the workbook.

Related Functions

PAGE.SETUP Sets page printing specifications

PRINT.PREVIEW Previews pages and page breaks before printing

PRINTER.SETUP Identifies the printer

SET.PRINT.AREA Defines the print area

SET.PRINT.TITLES Defines text to print as titles

DEFINE.NAME Equivalent to clicking the Define command on the Name submenu of the Insert menu

PRINTER.SETUP

Use PRINTER.SETUP to change the printer you are using.

Syntax

PRINTER.SETUP(**printer_text**)

PRINTER.SETUP?(printer_text)

Printer_text is the name of the printer you want to switch to. Enter printer_text exactly as it appears in the Setup dialog box.

Note This function is only available in Microsoft Excel for Windows.

Related Functions

PAGE.SETUP	Sets page printing specifications
PRINT	Prints the active workbook

PRINT.PREVIEW

Equivalent to clicking the Print Preview command on the File menu. Previews the pages and page breaks of the active workbook on the screen before printing.

Syntax

PRINT.PREVIEW( )

Related Function

PRINT	Prints the active workbook

PROMOTE

Equivalent to clicking the Ungroup button. Promotes, or ungroups, the currently selected rows or columns in an outline. Use PROMOTE to change the configuration of an outline by promoting rows or columns of information.

Syntax

PROMOTE(rowcol)

PROMOTE?(rowcol)

Rowcol specifies whether to promote rows or columns.

Rowcol	**Demotes**
1 or omitted	Rows
2	Columns

Remarks

- If the selection consists of an entire row or rows, then rows are promoted even if rowcol is 2. Similarly, selection of an entire column overrides rowcol 1.
- Also, if the selection is unambiguous (an entire row or column), then PROMOTE? will not display the dialog box.

Related Functions

DEMOTE	Demotes the selection in an outline
SHOW.DETAIL	Expands or collapses a portion of an outline
SHOW.LEVELS	Displays a specific number of levels of an outline

PROTECT.DOCUMENT

Adds protection to or removes protection from the active sheet, macro sheet, chart, dialog sheet, module, or scenario. Use PROTECT.DOCUMENT to prevent yourself or others from changing cell contents, or objects in a workbook. To protect workbooks in Microsoft Excel version 5.0 or later, see WORKBOOK.PROTECT.

Syntax

PROTECT.DOCUMENT(contents, windows, password, objects, scenarios)

PROTECT.DOCUMENT?(contents, windows, password, objects, scenarios)

Contents is a logical value corresponding to the Contents check box in the Protect Sheet dialog box.

- If contents is TRUE or omitted, Microsoft Excel selects the check box and protects cells and chart elements on the sheet or macro sheet.
- If contents is FALSE, Microsoft Excel clears the check box (and removes protection if the correct password is supplied).

Windows is provided for compatibility with Microsoft Excel version 4.0. To protect the window placement and structure of workbooks in Microsoft Excel version 5.0 or later, see WORKBOOK.PROTECT.

- If windows is TRUE, Microsoft Excel prevents a workbook's windows from being moved or sized.
- If windows is FALSE or omitted, Microsoft Excel removes protection if the correct password is supplied.

Password is the password you specify in the form of text to protect or unprotect the file. Password is case-sensitive.

- If password is omitted when you protect a sheet, then you will be able to remove protection without a password. This is useful if you want only to protect the sheet from accidental changes.

- If password is omitted when you try to remove protection from a sheet that was protected with a password, the normal password dialog box is displayed.

- Passwords are not recorded into the PROTECT.DOCUMENT function when you use the macro recorder.

Objects   is a logical value. This argument applies only to charts, worksheets and macro sheets. Objects corresponds to the Objects check box in the Protect Sheet dialog box.

- If objects is TRUE or omitted, Microsoft Excel selects the check box and protects all locked objects on the chart, worksheet or macro sheet.

- If objects is FALSE, Microsoft Excel clears the check box.

Scenarios   is a logical value that corresponds to the Scenarios check box on the Protect Sheet dialog box. If TRUE, Microsoft Excel protects all the scenarios. If FALSE, the scenarios are not protected.

Remarks

- If contents and objects are FALSE, PROTECT.DOCUMENT carries out the Unprotect Sheet command. If contents, or objects is TRUE, it carries out the Protect Sheet command.

- Make sure that you hide macro sheets that protect or unprotect worksheets. If you type a password directly into the function on an unhidden macro sheet, then someone could see the password needed to unprotect the worksheet. For example, `PROTECT.DOCUMENT(TRUE, TRUE, "XD1411C", TRUE).`

Warning   If you forget the password of a workbook that was previously protected with a password, you cannot unprotect the workbook.

Related Functions

CELL.PROTECTION   Controls protection for the selected cells

ENTER.DATA   Turns Data Entry mode on and off

OBJECT.PROTECTION   Controls how an object is protected

SAVE.AS   Saves a workbook and allows you to specify the name, file type, password, backup file, and location of the workbook

WORKBOOK.PROTECT   Protects a workbook

PTTESTM

Performs a paired two-sample Student's t-Test for means.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

PTTESTM(**inprng1, inprng2**, outrng, labels, alpha, difference)

PTTESTM?(inprng1, inprng2, outrng, labels, alpha, difference)

Inprng1 is the input range for the first data set.

Inprng2 is the input range for the second data set.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Labels is a logical value.

- If labels is TRUE, then labels are in the first row or column of the input ranges.
- If labels is FALSE or omitted, all cells in inprng1 and inprng2 are considered data. The output table will include default row or column headings.

Alpha is the confidence level for the test. If omitted, alpha is 0.05.

Difference is the hypothesized mean difference. If omitted, difference is 0.

Related Functions

PTTESTV Performs a two-sample Student's t-Test, assuming unequal variances

TTESTM Performs a two-sample Student's t-Test for means, assuming equal variances

PTTESTV

Performs a two-sample Student's t-Test, assuming unequal variances.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

PTTESTV(**inprng1, inprng2**, outrng, labels, alpha)

PTTESTV?(inprng1, inprng2, outrng, labels, alpha)

Inprng1 is the input range for the first data set.

Inprng2 is the input range for the second data set.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Labels is a logical value.

- If labels is TRUE, then labels are in the first row or column of the input ranges.
- If labels is FALSE or omitted, all cells in inprng1 and inprng2 are considered data. The output table will include default row or column headings.

Alpha is the confidence level for the test. If omitted, alpha is 0.05.

Related Functions

PTTESTM Performs a paired two-sample Student's t-Test for means

TTESTM Performs a two-sample Student's t-Test for means, assuming equal variances

PUSHBUTTON.PROPERTIES

Sets the properties of the push button control on a worksheet or dialog sheet.

Syntax

PUSHBUTTON.PROPERTIES(default_logical, cancel_logical, dismiss_logical, help_logical, accel_text, accel_text2)

PUSHBUTTON.PROPERTIES?(default_logical, cancel_logical, dismiss_logical, help_logical, accel_text, accel_text2)

Default_logical is a logical value that determines whether the button is the default button for the dialog. If TRUE, the button is the default button. If FALSE, the button is not the default button for the control.

Cancel_logical is a logical value that determines whether the button is activated when the dialog is cancelled with the Close button or the ESC key. If TRUE, the button is activated when the dialog box is cancelled, and edit boxes are not checked to see if they contain valid data types. If FALSE, the button is not activated when the dialog box is cancelled.

Dismiss_logical is a logical value that determines whether the button dismisses the dialog when pressed, as when the user presses the box's OK button. If TRUE, the button dismisses the dialog box. If FALSE, the button does not dismiss the dialog box.

Help_logical is a logical value that determines whether the button is activated when the user presses the F1 key. If TRUE, the button is activated when the user presses the F1 key. If FALSE, the button is not activated when the user presses the F1 key.

Accel_text is a text string containing the character to use as the dialog button's accelerator key. The character is matched against the text of the control, and the first matching character is underlined. When the user presses ALT+accel_text in Microsoft Excel for Windows or COMMAND+accel_text in Microsoft Excel for the Macintosh, the control is clicked. This argument is ignored for push button controls on worksheets.

Accel_text2 is a text string containing the second accelerator key on a dialog sheet. This argument is for only Far East versions of Microsoft Excel.

Related Functions

CHECKBOX.PROPERTIES	Sets various properties of check box and option box controls
SCROLLBAR.PROPERTIES	Sets the properties of the scroll bar and spinner controls
EDITBOX.PROPERTIES	Sets the properties of an edit box on a worksheet or dialog sheet

QUERY.GET.DATA

Builds a new query using the supplied information. The application Microsoft Query nor any dialog boxes are displayed.

Syntax

QUERY.GET.DATA(**connection_string**, **query_text**, keep_query_def, field_names, row_numbers, destination)

QUERY.GET.DATA?(connection_string, query_text, keep_query_def, field_names, row_numbers, destination)

Connection_string supplies information, such as the data source name, user ID, and passwords, necessary to making a SQL connection to an external data source. For example: "DSN=Myserver; Server=server1; UID=dbayer; PWD=buyer1; Database=nwind".

- You must define the data source name (DSN) used in connection_string before you try to connect to it.
- You can enter connection_string as an array or a string. If connection_string exceeds 250 characters, you must enter it as an array.
- If QUERY.GET.DATA is unable to access the data source using connection_string, it returns the #N/A error value.

Query_text is the SQL language query to be executed on the data source.

Keep_query_def is a logical value that, if TRUE or omitted, preserves the query definition. If FALSE, the query definition is lost and the data from the query no longer constitutes a data range.

Field_names is a logical value that, if TRUE or omitted, places field names from Microsoft Query into the first row of the data range. If FALSE, the field names are discarded.

Row_numbers is a logical value that, if TRUE, places row numbers from Microsoft Query into the first column in the data range. If FALSE or omitted, the row numbers are discarded.

Destination is the location as a cell reference where you want the data placed. If destination is in a current data range then that data range is changed to reflect the new SQL statement. The default destination is the currently selected cell or range.

Remarks

- If the information provided is not sufficient to create the query then the error value #REF! is returned.
- If Microsoft Query is unavailable or can not be found, #N/A is returned.
- If connection string is longer than 255 characters, the string will be truncated at the last semi-colon.

Related Function

QUERY.REFRESH Refreshes the data in a data range returned by Microsoft Query

QUERY.REFRESH

Refreshes the data in a data range returned to a worksheet from Microsoft Query. This function is equivalent to the Refresh button on the External Data toolbar.

Syntax

QUERY.REFRESH(reference)

Reference is the reference to a single cell inside a data range. If reference is not in a data range then the error value #REF! is returned.

Related Function

QUERY.GET.DATA   Builds a new query using the supplied information

QUIT

Equivalent to clicking the Exit command on the File menu in Microsoft Excel for Windows. Equivalent to clicking the Quit command on the File menu in Microsoft Excel for the Macintosh. Quits Microsoft Excel and closes any open workbooks. If open workbooks have unsaved changes, Microsoft Excel displays a message asking if you want to save them. You can use QUIT in an Auto_Close macro to force Microsoft Excel to quit when a particular workbook is closed.

Syntax

QUIT( )

Caution   If you have cleared error-checking with an ERROR(FALSE) function, QUIT will not ask whether you want to save changes.

Remarks

When you use the QUIT function, Microsoft Excel does not run any Auto_Close macros before closing the workbook.

Examples

The following function displays a confirmation alert and quits Microsoft Excel if the user clicks OK:

```
IF(ALERT("Are you sure you want to quit Microsoft Excel?",1), QUIT(),)
```

Related Function

FILE.CLOSE   Closes the active workbook

RANDOM

Fills a range with independent random or patterned numbers drawn from one of several distributions.

If this function is not available, you must install the Analysis ToolPak add-in.

RANDOM provides six different random distributions and one patterned data option. Because the distributions require different argument lists, there are seven syntax forms for RANDOM.

Syntax 1

Uniform distribution

RANDOM(outrng, variables, points, **distribution**, seed, **from, to**)

RANDOM?(outrng, variables, points, distribution, seed, from, to)

Syntax 2

Normal distribution

RANDOM(outrng, variables, points, **distribution**, seed, **mean, standard_dev**)

RANDOM?(outrng, variables, points, distribution, seed, mean, standard_dev)

Syntax 3

Bernoulli distribution

RANDOM(outrng, variables, points, **distribution**, seed, **probability**)

RANDOM?(outrng, variables, points, distribution, seed, probability)

Syntax 4

Binomial distribution

RANDOM(outrng, variables, points, **distribution**, seed, **probability, trials**)

RANDOM?(outrng, variables, points, distribution, seed, probability, trials)

Syntax 5

Poisson distribution

RANDOM(outrng, variables, points, **distribution**, seed, **lambda**)

RANDOM?(outrng, variables, points, distribution, seed, lambda)

Syntax 6

Patterned distribution

RANDOM(outrng, variables, points, **distribution**, seed, **from, to, step, repeat_num, repeat_seq**)

RANDOM?(outrng, variables, points, distribution, seed, from, to, step, repeat_num, repeat_seq)

Syntax 7

Discrete distribution

RANDOM(outrng, variables, points, **distribution**, seed, **inprng**)

RANDOM?(outrng, variables, points, distribution, seed, inprng)

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Variables is the number of random number sets to generate. RANDOM will generate variables columns of random numbers. If omitted, variables is equal to the number of columns in the output range.

Points is the number of data points per random number set. RANDOM will generate points rows of random numbers for each random number set. If omitted, points is equal to the number of rows in the output range. Points is ignored when distribution is 6 (Patterned).

Distribution indicates the type of number distribution.

Distribution	Distribution type
1	Uniform
2	Normal
3	Bernoulli

4	Binomial
5	Poisson
6	Patterned
7	Discrete

Seed   is an optional value with which to begin random number generation. Seed is ignored when distribution is 6 (Patterned) or 7 (Discrete).

From   is the lower bound.

To   is the upper bound.

Mean   is the mean.

Standard_dev   is the standard deviation.

Probability   is the probability of success on each trial.

Trials   is the number of trials.

Lambda   is the Poisson distribution parameter.

Step   is the increment between from and to.

Repeat_num   is the number of times to repeat each value.

Repeat_seq   is the number of times to repeat each sequence of values.

Inprng   is a two-column range of values and their probabilities.

RANKPERC

Returns a table that contains the ordinal and percent rank of each value in a data set.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

RANKPERC(**inprng**, outrng, grouped, labels)

RANKPERC?(inprng, outrng, grouped, labels)

Inprng   is the input range.

Outrng   is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Grouped   is a text character that indicates whether the data in the input range is organized by row or column.

- If grouped is "C" or omitted, then the data is organized by column.
- If grouped is "R", then the data is organized by row.

Labels is a logical value that describes where the labels are located in the input range, as shown in the following table:

Labels	Grouped	Labels are in
TRUE	"C"	First row of the input range.
TRUE	"R"	First column of the input range.
FALSE or omitted	(ignored)	No labels. All cells in the input range are data.

REFTEXT

Converts a reference to an absolute reference in the form of text. Use REFTEXT when you need to manipulate references with text functions. After manipulating the reference text, you can convert it back into a normal reference by using TEXTREF.

Syntax

REFTEXT(**reference**, a1)

Reference is the reference you want to convert.

A1 is a logical value specifying A1-style or R1C1-style references.

- If a1 is TRUE, REFTEXT returns an A1-style reference.
- If a1 is FALSE or omitted, REFTEXT returns an R1C1-style reference.

Examples

`REFTEXT(C3, TRUE)` equals "$C$3"

`REFTEXT(B2:F2)` equals "R2C2:R2C6"

If the active cell is B9 on the active sheet named SHEET1, then:

`REFTEXT(ACTIVE.CELL())` equals "[Book1]SHEET1!R9C2"

`REFTEXT(ACTIVE.CELL(), TRUE)` equals "[Book1]SHEET1!$B$9"

Related Functions

ABSREF Returns the absolute reference of a range of cells to another range

DEREF Returns the values of cells in the reference

RELREF Returns a relative reference

TEXTREF Converts text to a reference

REGISTER

Registers the specified dynamic link library (DLL) or code resource and returns the register ID. You can also specify a custom function name and argument names that will appear in the Paste Function dialog box. If you register a command (macro_type = 2), you can also specify a shortcut key. Because Microsoft Excel for Windows and Microsoft Excel for the Macintosh use different types of code resources, REGISTER has a slightly different syntax form when used in each operating environment.

Important This function is provided for advanced users only. If you use the CALL function incorrectly, you could cause errors that will require you to restart your computer.

Syntax 1

For Microsoft Excel for Windows

REGISTER(**module_text**, procedure, type_text, function_text, argument_text, macro_type, category, shortcut_text, help_topic, function_help, argument_help1, argument_help2,...)

Syntax 2

For Microsoft Excel for the Macintosh

REGISTER(**file_text**, resource, type_text, function_text, argument_text, macro_type, category, shortcut_text, help_topic, function_help, argument_help1, argument_help2,...)

Module_text or file_text is text specifying the name of the DLL that contains the function (in Microsoft Excel for Windows) or the name of the file that contains the code resource (in Microsoft Excel for the Macintosh).

Procedure or resource is text specifying the name of the function in the DLL (in Microsoft Excel for Windows) or the name of the code resource (in Microsoft Excel for the Macintosh). In Microsoft Excel for Windows, you can also use the ordinal value of the function from the EXPORTS statement in the module-definition file (.DEF). In Microsoft Excel for the Macintosh, you can also use the resource ID number. The ordinal value or resource ID number should not be in text form.

This argument may be omitted for stand-alone DLLs or code resources. In this case, REGISTER will register all functions or code resources and then return module_text or file_text.

Type_text is text specifying the data type of the return value and the data types of all arguments to the DLL or code resource. The first letter of type_text specifies the return value.

Function_text is text specifying the name of the function as you want it to appear in the Paste Function dialog box. If you omit this argument, the function will not appear in the Paste Function dialog box.

Argument_text is text specifying the names of the arguments you want to appear in the Paste Function dialog box. Argument names should be separated by commas.

Macro_type specifies the macro type: 1 for a function or 2 for a command. If macro_type is omitted, it is assumed to be 1 (function).

Category specifies the function category in the Paste Function dialog box in which you want the registered function to appear. You can use the category number or the category name for category. If you use the category name, be sure to enclose it in double quotation marks. If category is omitted, it is assumed to be 14 (User Defined).

Category number	Category name
1	Financial
2	Date & Time
3	Math & Trig
4	Text
5	Logical
6	Lookup & Matrix
7	Database
8	Statistical
9	Information
10	Commands (macro sheets only)
11	Actions (macro sheets only)
12	Customizing (macro sheets only)
13	Macro Control (macro sheets only)
14	User Defined

Shortcut_text is a character specifying the shortcut key for the registered command. The shortcut key is case-sensitive. This argument is used only if macro_type = 2 (command). If shortcut_text is omitted, the command will not have a shortcut key.

Help_topic is the reference (including path) to the help file that you want displayed when the user clicks the Help button when your custom function is displayed.

Function_help is a text string describing your custom function when it is selected in the Paste Function dialog box. The maximum number of characters is 255.

Argument_help1, argument_help2 are 1 to 21 text strings that describes you custom function's arguments when the function is selected in the Paste Function dialog box.

Example

Syntax 1

In Microsoft Excel for Windows, the following macro formula registers the GetTickCount function from Microsoft Windows. This function returns the number of milliseconds that have elapsed since Microsoft Windows was started.

```
REGISTER("User", "GetTickCount", "J")
```

Assuming that the REGISTER function is in cell A5, after your macro registers GetTickCount, you can use the CALL function to return the number of milliseconds that have elapsed:

```
CALL(A5)
```

Example

Syntax 1 with optional function_text

You can use the following macro formula to register the GetTickCount function from Microsoft Windows and assign the custom name GetTicks to it. To do this, include "GetTicks" as the optional function_text argument to the REGISTER function.

```
REGISTER("User", "GetTickCount", "J", "GetTicks", , 1, 9)
```

After the function is registered, the custom name GetTicks will appear in the Information function category (category = 9) in the Paste Function dialog box.

You can call the function from the same macro sheet on which it was registered using the following formula:

```
GetTicks()
```

You can call the function from another sheet or macro sheet by including the name of the original macro sheet in the formula. For example, assuming the macro sheet on which GetTicks was registered is named MACRO1.XLS, the following formula calls the function from another sheet:

```
MACRO1.XLS!GetTicks()
```

Tip You can use functions in a DLL or code resource directly on a sheet without first registering them from a macro sheet. Use syntax 2a or 2b of the CALL function. For more information, see CALL.

Related Function

UNREGISTER Removes a registered code resource from memory

REGRESS

Performs multiple linear regression analysis.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

REGRESS(**inpyrng, inpxrng**, constant, labels, confid, soutrng, residuals, sresiduals, rplots, lplots, routrng, nplots, poutrng)

REGRESS?(inpyrng, inpxrng, constant, labels, confid, soutrng, residuals, sresiduals, rplots, lplots, routrng, nplots, poutrng)

Inpyrng   is the input range for the y-values (dependent variable).

Inpxrng   is the input range for the x-values (independent variable).

Constant   is a logical value. If constant is TRUE, the y-intercept is assumed to be zero (the regression line passes through the origin). If constant is FALSE or omitted, the y-intercept is assumed to be a non-zero number.

Labels   is a logical value.

- If labels is TRUE, then the first row or column of the input ranges contain labels.
- If labels is FALSE or omitted, all cells in inpyrng and inpxrng are considered data. Microsoft Excel will then generate the appropriate data labels for the output table.

Confid   is an additional confidence level to apply to the regression. If omitted, confid is 95%.

Soutrng   is the first cell (the upper-left cell) in the output table or the name, as text, of the new sheet to contain the summary output table. If FALSE, blank, or omitted, places the summary output table in a new workbook. Microsoft Excel version 5.0 uses a single output table for REGRESS; Microsoft Excel version 4.0 used three different output tables for summary, residual, and probability data.

Residuals   is a logical value. If residuals is TRUE, REGRESS includes residuals in the output table. If residuals is FALSE or omitted, residuals are not included.

Sresiduals   is a logical value. If sresiduals is TRUE, REGRESS includes standardized residuals in the output table. If sresiduals is FALSE or omitted, standardized residuals are not included.

Rplots   is a logical value. If rplots is TRUE, REGRESS generates separate charts for each x versus the residual. If rplots is FALSE or omitted, separate charts are not generated.

Lplots   is a logical value. If lplots is TRUE, REGRESS generates a chart showing the regression line fitted to the observed values. If lplots is FALSE or omitted, the chart is not generated.

Routrng   is the first cell (the upper-left cell) in the residuals output table or the name, as text, of the new sheet to contain the residuals output table. If FALSE, blank, or omitted, places the residuals output table in a new worksheet. This argument is for compatibility with Microsoft Excel version 4.0 only and is ignored in later versions.

Nplots   is a logical value. If nplots is TRUE, REGRESS generates a chart of normal probabilities. If nplots is FALSE or omitted, the chart is not generated.

Poutrng   is the first cell (the upper-left cell) in the probability data output table or the name, as text, of the new sheet to contain the probability output table. If FALSE, blank, or omitted, places the probability output table in a new worksheet. This argument is for compatibility with Microsoft Excel version 4.0 only and is ignored in later versions.

RELREF

Returns the reference of a cell or cells relative to the upper-left cell of rel_to_ref. The reference is given as an R1C1-style relative reference in the form of text, such as "R[1]C[1]".

Syntax

RELREF(**reference, rel_to_ref**)

Reference is the cell or cells to which you want to create a relative reference.

Rel_to_ref is the cell from which you want to create the relative reference.

Tip If you know the absolute reference of a cell that you want to include in a formula, but your formula requires a relative reference, use RELREF to generate the relative reference. This is especially useful with the FORMULA function, since its formula_text argument requires R1C1-style references, and RELREF returns relative R1C1-style references. You can also use the FORMULA.CONVERT function to convert absolute references to relative references.

Examples

`RELREF($A$1, $C$3)` equals "R[-2]C[-2]"

`RELREF($A$1:$E$5, $C$3:$G$7)` equals "R[-2]C[-2]:R[2]C[2]"

`RELREF($A$1:$E$5, $C$3)` equals "R[-2]C[-2]:R[2]C[2]"

Related Functions

ABSREF Returns the absolute reference of a range of cells to another range

DEREF Returns the value of the cells in the reference

FORMULA Enters values into a cell or range or onto a chart

FORMULA.CONVERT Changes the reference style and type

REMOVE.LIST.ITEM

Removes an item in a list box or drop-down box.

Syntax

REMOVE.LIST.ITEM(**index_num**, count_num)

Index_num specifies the index of the item to remove, from 1 to the number of items in the list. Specify zero to remove all items in the list.

Count_num Specifies the number of items to delete starting from index_num. If omitted, only one item is removed.

Remarks

If count_num + index_num is greater than the number of items in the list, all items starting with index_num to the end of the list are removed.

Examples

`REMOVE.LIST.ITEM(3,2)` removes two items starting with the third item

`REMOVE.LIST.ITEM(3)` removes only the third item

Related Function

LISTBOX.PROPERTIES   Sets the properties of a list box and drop-down controls on worksheet and dialog sheets

REMOVE.PAGE.BREAK

Equivalent to clicking the Remove Page Break command on the Insert menu. Removes manual page breaks that you set with the SET.PAGE.BREAK function or the Page Break command on the Insert menu. If the active cell is not below or to the right of a manual page break, REMOVE.PAGE.BREAK takes no action. If the entire sheet is selected, REMOVE.PAGE.BREAK removes all manual page breaks. REMOVE.PAGE.BREAK does not remove automatic page breaks.

Syntax

REMOVE.PAGE.BREAK( )

Related Function

SET.PAGE.BREAK   Sets manual page breaks

RENAME.COMMAND

Changes the name of a built-in or custom menu command or the name of a menu. Use RENAME.COMMAND to change the name of a command on a menu, for example, when you create two custom commands that toggle on the menu. Examples of two built-in commands that toggle are the Page Break and Remove Page Break commands on the Insert menu.

Syntax

RENAME.COMMAND(**bar_num, menu, command, name_text**, position)

Bar_num   can be either the number of one of the Microsoft Excel built-in menu bars or the number returned by a previously run ADD.BAR function. See ADD.COMMAND for a list of ID numbers for built-in menu bars.

Menu   can be either the name of a menu as text or the number of a menu. Menus are numbered starting with 1 from the left of the screen.

Command   can be either the name of the command as text or the number of the command to be renamed (the first command on a menu is command 1). If command is 0, RENAME.COMMAND renames the menu instead of the command. Because other macros can change the position of custom menu commands, you should use the name of the command rather than a number whenever possible.

If the specified menu bar, menu, or command does not exist, RENAME.COMMAND returns the #VALUE! error value and interrupts the macro.

Name_text   is the new name for the command.

Position   is the name of the command on a submenu that you want to rename. If you use position, you must use command as the name of the submenu.

Tip   To specify an access key for the new name, precede the character you want to use with an ampersand (&). The access key is indicated by an underline under one letter of a menu or command name. In Microsoft Excel for the Macintosh, you can use the General tab in the Options dialog box to turn command underlining on or off. To see the Options dialog box, click Options on the Tools menu.

Example

To rename the Save All command as Global Save, and to make the letter "G" in Global Save an access key, use the following macro formula:

```
RENAME.COMMAND(10, "File", "Save All", "&Global Save")
```

Related Functions

ADD.BAR	Adds a menu bar
ADD.COMMAND	Adds a command to a menu
CHECK.COMMAND	Adds or deletes a check mark to or from a command
DELETE.COMMAND	Deletes a command from a menu
ENABLE.COMMAND	Enables or disables a menu or custom command

RENAME.OBJECT

Renames the selected object or group. This is useful for giving objects names more relevant to their usage. This is also useful if it is uncertain how the object may have been named.

Syntax

RENAME.OBJECT(new_name)

New_name is the new name to be given to the selected object.

Related Functions

GET.OBJECT	Returns information about a specified object
INSERT.OBJECT	Equivalent to clicking the Object command on the Insert menu
SELECT Syntax 2	Selects objects on worksheets

REPLACE.FONT

Replaces one of the four built-in fonts in Microsoft Excel for Windows version 2.1 or earlier with a new font and style. This function is included only for macro compatibility. To change the font of the selected cell or range as part of a macro, use the FONT.PROPERTIES function.

Syntax

REPLACE.FONT(font_num, name_text, size_num, bold, italic, underline, strike, color, outline, shadow)

Related Function

FONT.PROPERTIES	Sets various font attributes

REPORT.DEFINE

Equivalent to clicking the Report Manager command on the View menu and then clicking the Add option in the Report Manager dialog box. Creates or replaces a report definition. If this function is not available, you must install the Report Manager add-in.

Syntax

REPORT.DEFINE(**report_name, sections_array**, pages_logical)

Report_name specifies the name of the report. If the workbook already contains a report with report_name, the new report replaces the existing one.

Sections_array is an array that contains one or more rows of view, scenario, and sheet name that define the report. The sheet name is the sheet on which the view and scenario are defined. If the sheet name is not specified, the current sheet is used when REPORT.DEFINE is run.

Pages_logical is a logical value that, if TRUE or omitted, specifies continuous page numbers for multiple sections or, if FALSE, resets page numbers to 1 for each new section.

Remarks

- REPORT.DEFINE returns the #VALUE error value if report_name is invalid or if the workbook is protected.
- If there are no reports defined, this function will bring up the Add Report dialog box.

Related Functions

REPORT.DELETE Removes a report from the active workbook

REPORT.PRINT Prints a report

REPORT.GET Returns information about reports defined for the active workbook

REPORT.DELETE

Equivalent to clicking the Report Manager command on the View menu and then selecting a report in the Report Manager dialog box and clicking the Delete button. Removes a report definition from the active workbook.

If this function is not available, you must install the Report Manager add-in.

Syntax

REPORT.DELETE(**report_name**)

Report_name specifies the name of the report to be removed. Report_name can be any text that does not contain quotation marks.

Remarks

REPORT.DELETE returns the #VALUE error value if report_name is invalid or if the workbook is protected.

Related Functions

REPORT.DEFINE Creates a report

REPORT.PRINT Prints a report

REPORT.GET Returns information about reports defined for the active workbook

REPORT.GET

Returns information about reports defined for the active workbook. Use REPORT.GET to return information you can use in other macro commands that manipulate reports.

If this function is not available, you must install the Report Manager add-in.

Syntax

REPORT.GET(**type_num**, report_name)

Type_num   is a number from 1 to 3 specifying the type of information to return, as shown in the following table.

Type_num	Returns
1	An array of reports from all sheets in the active workbook or the #N/A error value if none are specified
2	An array of views, scenarios, and sheet names for the specified report in the active workbook. REPORT.GET returns the #N/A error value if the scenario check box is not selected. Returns the #VALUE! error value if name is invalid or the workbook is protected.
3	If continuous page numbers are used, returns TRUE. If page numbers start at 1 for each section, returns FALSE. Returns the #VALUE! error value if report_name is invalid or the workbook is protected.

Report_name   specifies the name of a report in the active workbook.

Remarks

Report_name is required if type_num is 2 or 3.

Example

The following macro formula returns an array of reports from the active workbook.

```
REPORT.GET(1)
```

Related Functions

REPORT.DEFINE   Creates a report

REPORT.DELETE   Removes a report from the active workbook

REPORT.PRINT   Prints a report

REPORT.PRINT

Equivalent to clicking the Print button in the Report Manager dialog box. Prints a report.

If this function is not available, you must install the Report Manager add-in.

Syntax

REPORT.PRINT(**report_name**, copies_num, show_print_dlg_logical)

REPORT.PRINT?(report_name, copies_num)

Report_name specifies the name of a report in the active workbook.

Copies_num is the number of copies you want to print. If omitted, the default is 1.

Show_print_dlg_logical is a logical value that, if TRUE, displays a dialog box asking how many copies to print, or, if FALSE or omitted, prints the report immediately using existing print settings.

Remarks

REPORT.PRINT returns the #VALUE! error value if report_name is invalid or if the workbook is protected.

Related Functions

REPORT.DEFINE Creates a report

REPORT.DELETE Removes a report from the active workbook

REQUEST

Requests an array of a specific type of information from an application with which you have a dynamic data exchange (DDE) link. Use REQUEST with other Microsoft Excel DDE functions to move information from another application into Microsoft Excel.

Syntax

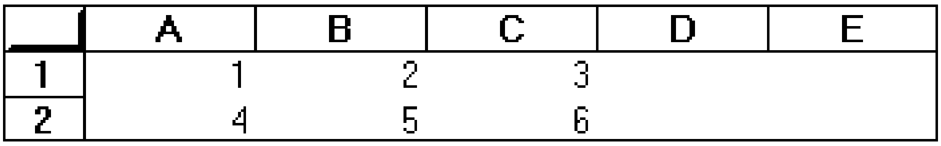
REQUEST(**channel_num, item_text**)

Important Microsoft Excel for the Macintosh requires system software version 7.0 or later for this function.

Channel_num is a number returned by a previously run INITIATE function. Channel_num refers to a channel through which Microsoft Excel communicates with another program.

Item_text is a code indicating the type of information you want to request from another application. The form of item_text depends on the application connected to channel_num.

REQUEST returns the data as an array. For example, suppose the remote data to be returned came from a sheet that looked like the following illustration.



	A	B	C	D	E
1	1	2	3		
2	4	5	6		

REQUEST would return that data as the following array:

{1, 2, 3;4, 5, 6}

If REQUEST is not successful, it returns the following error values.

Value returned	Situation
#VALUE!	Channel_num is not a valid channel number.
#N/A	The application you are accessing is busy doing something else.
#DIV/0!	The application you are accessing does not respond after a certain length of time, or you have pressed ESC or COMMAND+PERIOD to cancel.
#REF!	The request is refused.

Tip Use the ERROR.TYPE function to distinguish between the different error values.

Example

Suppose you had opened a DDE channel to Microsoft Word for Windows. WChan contains the number of the open channel. In Microsoft Excel for Windows, the following function returns the text specified by the bookmark named BMK1.

```
=REQUEST(WChan, "BMK1")
```

Related Functions

EXECUTE   Carries out a command in another application

INITIATE   Opens a channel to another application

POKE   Sends data to another application

SEND.KEYS   Sends a key sequence to another application

TERMINATE   Closes a dynamic data exchange (DDE) channel previously opened with the INITIATE function

RESET.TOOL

Resets a button to its original button face.

Syntax

RESET.TOOL(**bar_id, position**)

Bar_id   is the number or name of the toolbar containing the button you want to reset. For detailed information about bar_id, see ADD.TOOL.

Position   specifies the position of the button within the toolbar. Position starts with 1 at the left side (if horizontal) or at the top (if vertical).

Related Functions

ADD.TOOL   Adds one or more buttons to a toolbar

DELETE.TOOL   Deletes a button from a toolbar

RESET.TOOLBAR   Resets a button to its original button face

RESET.TOOLBAR

Resets built-in toolbars to the default Microsoft Excel set.

Syntax

RESET.TOOLBAR(bar_id)

Bar_id   specifies the number or name of the toolbar that you want to reset. For detailed information about bar_id, see ADD.TOOL.

Remarks

If RESET.TOOLBAR successfully resets the toolbar, it returns TRUE. If you try to reset a custom toolbar, RESET.TOOLBAR returns #VALUE! and takes no other action.

Related Functions

ADD.TOOL   Adds one or more tools to a toolbar

DELETE.TOOLBAR   Deletes custom toolbars

RESTART

Removes a number of RETURN statements from the stack. When one macro calls another, the RETURN statement at the end of the second macro returns control to the calling macro. You can use the RESTART function to determine which macro regains control.

Syntax

RESTART(level_num)

Level_num   is a number specifying the number of previous RETURN statements you want to be ignored. If level_num is omitted, the next RETURN statement will halt macro execution.

For example, if the currently running macro has two "ancestors" (the current macro was called by one macro that, in turn, was called by another macro), using RESTART(1) in the third macro returns control to the first calling macro when the RETURN statement is encountered. The RESTART(1) formula removes one level of RETURN statements from Microsoft Excel's memory so that the second macro is skipped.

Remarks

RESTART is particularly useful if you frequently use macros to call other macros that in turn call other macros. Use RESTART in combination with IF statements to prevent macro execution from returning to macros that called, either directly or indirectly, the currently running macro.

Related Functions

HALT   Stops all macros from running

RETURN   Ends the currently running macro

RESULT

Specifies the type of data a macro or custom function returns. Use RESULT to make sure your macros, custom functions, or subroutines return values of the correct data type.

Syntax

RESULT(type_num)

Type_num is a number specifying the data type.

Type_num	Type of returned data
1	Number
2	Text
4	Logical
8	Reference
16	Error
64	Array

- Type_num can be the sum of the numbers in the preceding table to allow for more than one possible result type. For example, if type_num is 12, which equals 4 + 8, the result can be a logical or a reference value.
- If you omit type_num, it is assumed to be 7. Since 7 equals 1 + 2 + 4, the value returned can be a number (1), text (2), or logical value (4).

Examples

The following function specifies that a custom function's return value can be a number or a logical value (4+1=5):

```
RESULT(5)
```

Related Functions

ARGUMENT Passes an argument to a macro

RETURN Ends the currently running macro

RESUME

Equivalent to choosing the Resume button on the toolbar. Resumes a paused macro. Returns TRUE if successful or the #VALUE! error value if no macro is paused. A macro can be paused by using the PAUSE function or choosing Pause from the Single Step dialog box, which appears when you choose the Step Into button from the Macro dialog box.

Syntax

RESUME(type_num)

Type_num is a number from 1 to 4 specifying how to resume.

Type_num	How Microsoft Excel resumes
1 or omitted	If paused by a PAUSE function, continues running the macro. If paused from the Single Step dialog box, returns to that dialog box.
2	Halts the paused macro
3	Continues running the macro
4	Opens the Single Step dialog box

Tip You can use Microsoft Excel's ON functions to resume based on an event. For an example, see ENTER.DATA.

Remarks

- If one macro runs a second macro that pauses, and you need to halt only the paused macro, use RESUME(2) instead of HALT. HALT halts all macros and prevents resuming or returning to any macro.
- If the macro was paused from the Single Step dialog box, RESUME returns to the Single Step dialog box.

Related Functions

HALT Stops all macros from running

PAUSE Pauses a macro

RETURN Ends the currently running macro

RETURN

Ends the currently running macro. If the currently running macro is a subroutine macro that was called by another macro, control is returned to the calling macro. If the currently running macro is a custom function, control is returned to the formula that called the custom function. If the currently running macro is a command macro started by the user with the Run button in the Macro dialog box or a shortcut key or by clicking an object, control is returned to the user.

Syntax

RETURN(value)

Value specifies what to return.

- If the macro is a custom function or a subroutine, value specifies what value to return. However, not all subroutines return values; the last line in macros that do not return values is =RETURN().
- If the macro is a command macro run by the user, value should be omitted.

Remarks

RETURN signals the end of a macro. Every macro must end with a RETURN or HALT function, but not every macro returns values.

Example

The following function returns the sum of the range B1:B10:

```
RETURN(SUM(B1:B10))
```

Related Functions

BREAK Interrupts a FOR-NEXT, FOR.CELL-NEXT, or WHILE-NEXT loop

HALT Stops all macros from running

RESULT Specifies the data type a custom function returns

ROUTE.DOCUMENT

Routes the workbook using the defined routing slip information.

Syntax

ROUTE.DOCUMENT()

Remarks

If there is no routing slip, returns #N/A. If an error occurs or routing is not enabled for the system, returns #VALUE!.

Related Functions

SEND.MAIL Sends the active workbook using email

ROUTING.SLIP Adds or Edits the routing slip attached to the current workbook

ROUTING.SLIP

Equivalent to clicking the Add Routing Slip command on the File menu. Adds or Edits the routing slip attached to the current workbook.

Syntax

ROUTING.SLIP(recipients,subject, message, route_num, return_logical, status_logical)

ROUTING.SLIP?(recipients,subject, message, route_num, return_logical, status_logical)

Recipients is the name of the person to whom you want to send the mail. The name should be given as text.

- To specify more than one name, give the list of names as an array. For example, ROUTING.SLIP({"John", "Paul", "George", "Ringo"}) would send the active workbook to the four names in the array. You can also refer to a range on a sheet or macro sheet that contains a list of names to whom you want the mail to be sent.

- Specifying recipients while a routing is in progress only modifies the non-grayed recipients (that is, those recipients who have not received the message yet). Recipients who have already received, reviewed and forwarded the routed workbook cannot be modified.

Subject is a text string containing the subject text used for the mail messages used to route the workbook. If omitted, the default subject line is "Routing: name", where name is the file name or title as displayed in the Summary Info dialog box, if available.

Message is a text string containing the body text used for the mail messages used to route the workbook.

Route_num is a number indicating the type of routing method.

Route_num	**Method**
1 or omitted	One after another routing
2	All at once routing

Return_logical is a logical value which, if TRUE or omitted, indicates that the routing should be returned to the originator when the routing is complete. If FALSE, the routing will end with the last recipient in the To list box in the Routing Slip Dialog box.

Status_logical is a logical value corresponding to the Track Status check box in the Routing Slip dialog box. If TRUE or omitted, status tracking messages for the routing are sent. FALSE means that no status tracking is performed.

Remarks

- If this function is used on a workbook that is already being routed, the route_num, status_logical and return_logical arguments are ignored (they cannot be changed).

- When arguments are omitted and a routing slip already exists, the omitted arguments are replaced by the current values of the routing slip.

Related Functions

ROUTE.DOCUMENT Routes the workbook using the defined routing slip information

SEND.MAIL Sends the active workbook using email

ROW.HEIGHT

Equivalent to choosing the Height command on the Row submenu of the Format menu. Changes the height of the rows in a reference.

Syntax

ROW.HEIGHT(height_num, reference, standard_height, type_num)

ROW.HEIGHT?(height_num, reference, standard_height, type_num)

Height_num specifies how high you want the rows to be in points. If standard_height is TRUE, height_num is ignored.

Reference specifies the rows for which you want to change the height.

- If reference is omitted, the reference is assumed to be the current selection.
- If reference is specified, it must be either an external reference to the active worksheet, such as !$2:$4 or !Database, or an R1C1-style reference in the form of text or a name, such as "R1:R3", "R[-4]:R[-2]", or Database.
- If reference is a relative R1C1-style reference in the form of text, it is assumed to be relative to the active cell.

Standard_height is a logical value that sets the row height as determined by the font in each row.

- If standard_height is TRUE, Microsoft Excel sets the row height to a standard height that may vary from row to row depending on the fonts used in each row, ignoring height_num.
- If standard_height is FALSE or omitted, Microsoft Excel sets the row height according to height_num.

Type_num is a number from 1 to 3 corresponding to selecting the Hide, Unhide, or AutoFit commands from the Row submenu.

Type_num	Action taken
1	Hides the row selection by setting the row height to 0
2	Unhides the row selection by setting the row height to the value set before the selection was hidden
3	Sets the row selection to an AutoFit height, which varies from row to row depending on how large the font is in any cell in each row or on how many lines of text are wrapped

Remarks

- If any of the argument settings conflict, such as when standard_height is TRUE and type_num is 3, Microsoft Excel uses the type_num argument and ignores any arguments that conflict with type_num.
- If you are recording a macro while using a mouse, and you change row heights by dragging the row border, Microsoft Excel records the reference of the rows using R1C1-style references in the form of text. If Uses Relative References is selected, Microsoft Excel uses R1C1-style relative references. If Uses Relative References is not selected, Microsoft Excel uses R1C1-style absolute references.

Related Function

COLUMN.WIDTH   Sets the widths of the specified columns

RUN

Equivalent to choosing the Run button in the Macro dialog box, which appears when you choose the Macros command on the Macro submenu of the Tools menu. Runs a macro.

Syntax

RUN(reference, step)

RUN?(reference, step)

Reference   is a reference to the macro you want to run or a number from 1 to 4 specifying an Auto macro to run.

If reference is	Specifies
1	All Auto_Open macros on the active workbook
2	All Auto_Close macros
3	All Auto_Activate macros
4	All Auto_Deactivate macros

- If reference is a range of cells, RUN begins with the macro function in the upper-left cell of reference.
- If the macro sheet containing the macro is not in the active workbook, reference can be an external reference to the name of the macro, such as `RUN([BOOK1]Macro!Months)` or an external R1C1-style reference to the location of the macro, such as `RUN("[Book1]Macro!R2C3")`. The reference must be in text form.

- If reference is omitted, the macro function in the active cell is carried out, and macro execution continues down that column.

Step is a logical value specifying that the macro is to be run in single-step mode. If step is TRUE, Microsoft Excel runs the macro in single-step mode; if FALSE or omitted, Microsoft Excel runs the macro normally.

Remarks

- RUN is recorded when you choose the Run button the Macro dialog box while recording a macro. The reference you enter in the Run dialog box is recorded as text, with A1-style references converted to R1C1-style references.
- To run a macro from a macro sheet, you could alternatively enter the name of the macro as a formula, followed by a set of parentheses. For example, enter `=[Book1]Macro!Months()` instead of `=RUN([Book1]Macro!Months)`.

Related Function

GOTO Directs macro execution to another cell

SAMPLE

Samples data.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

SAMPLE(**inprng**, outrng, **method, rate**, labels)

SAMPLE?(inprng, outrng, method, rate, labels)

Inprng is the input range.

Outrng is the first cell (the upper-left cell) in the output column or the name, as text, of a new sheet to contain the output column. If FALSE, blank, or omitted, places the output table in a new workbook.

Method is a text character that indicates the type of sampling.

- If method is "P", then periodic sampling is used. The input range is sampled every nth cell, where n = rate.
- If method is "R", then random sampling is used. The output column will contain rate samples.

Rate is the sampling rate, if method is "P" (periodic sampling). Rate is the number of samples to take if method is "R" (random sampling).

Labels is a logical value.

- If labels is TRUE, then the first row or column of inprng contains labels.

- If labels is FALSE or omitted, all cells in inprng are considered data. Microsoft Excel generates appropriate data labels for the output table.

SAVE

Equivalent to choosing the Save command from the File menu. Saves the active workbook.

Syntax

SAVE( )

Remarks

Use the SAVE.AS function instead of SAVE when you want to change the filename or file type, specify a password, create a backup file, or save a file to a different directory or folder.

Related Functions

SAVE.AS   Saves a workbook and allows you to specify the name, file type, password, backup file, and location of the workbook

SAVE.WORKBOOK   Saves a workbook

SAVE.AS

Equivalent to clicking the Save As command on the File menu. Use SAVE.AS to specify a new filename, file type, protection password, or write-reservation password, or to create a backup file.

Syntax

SAVE.AS(document_text, type_num, prot_pwd, backup, write_res_pwd, read_only_rec)

SAVE.AS?(document_text, type_num, prot_pwd, backup, write_res_pwd, read_only_rec)

Document_text   specifies the name of a workbook to save, such as SALES.XLS (in Microsoft Excel for Windows) or SALES (in Microsoft Excel for the Macintosh). You can include a full path in document_text, such as C:\EXCEL\ANALYZE.XLS (in Microsoft Excel for Windows) or HARDDISK:FINANCIALS:ANALYZE (in Microsoft Excel for the Macintosh).

Type_num   is a number specifying the file format in which to save the workbook.

Type_num	File format
1 or omitted	Normal
2	SYLK
3	Text
4	WKS
5	WK1

6	CSV
7	DBF2
8	DBF3
9	DIF
10	Reserved
11	DBF4
12	Reserved
13	Reserved
14	Reserved
15	WK3
16	Microsoft Excel 2.x
17	Template
18	Add-in macro (For compatibility only. In Microsoft Excel version 5.0, this saves as normal.)
19	Text (Macintosh)
20	Text (Windows)
21	Text (MS-DOS)
22	CSV (Macintosh)
23	CSV (Windows)
24	CSV (MS-DOS)
25	International macro
26	International add-in macro

27	Reserved
28	Reserved
29	Microsoft Excel 3.0
30	WK1 / FMT
31	WK1 / Allways
32	WK3 / FM3
33	Microsoft Excel 4.0
34	WQ1
35	Microsoft Excel 4.0 workbook
36	Formatted text (space delimited)

The following table shows which values of type_num apply to the six Microsoft Excel document types.

Document Type	**Type_num**
Worksheet	All except 10, 12-14, 18, 25-28, 36
Chart sheet	All except 10, 12-14, 18, 25-28
Visual Basic module	1, 3, 17
Dialog	1, 17
Macro sheet	1-3, 6, 9, 16-29, 33
Workbook	1, 15, 35

Prot_pwd corresponds to the Protection Password box in the Save Options dialog box in Microsoft Excel 95 or earlier versions, or the Password To Open box in Microsoft Excel 97 or later.

- Prot_pwd is a password given as text or as a reference to a cell containing text. Prot_pwd should be no more than 15 characters.

- If a file is saved with a password, the password must be supplied for the file to be opened.

Backup is a logical value corresponding to the Always Create Backup check box in the Save Options dialog box and specifies whether to make a backup workbook. If backup is TRUE, Microsoft Excel creates a backup file; if FALSE, no backup file is created; if omitted, the status is unchanged.

Write_res_pwd corresponds to the Write Reservation Password box in the Save Options dialog box in Microsoft Excel 95 or earlier versions, or the Password To Modify box in Microsoft Excel 97 or later. Allows the user to write to a file. If a file is saved with a password and the password is not supplied when the file is opened, the file is opened read-only.

Read_only_rec is a logical value corresponding to the Read-Only Recommended check box in the Save Options dialog box.

- If read_only_rec is TRUE, Microsoft Excel saves the workbook as a read-only recommended workbook; if FALSE, Microsoft Excel saves the workbook normally; if omitted, Microsoft Excel uses the current settings.

- When you open a workbook that was saved as read-only recommended, Microsoft Excel displays a message recommending that you open the workbook as read-only.

Related Functions

CLOSE Closes the active window

GET.DOCUMENT Returns information about a workbook

SAVE Saves the active workbook

SAVE.WORKBOOK Saves a workbook

SAVE.COPY.AS

Saves a copy of the current workbook using a different name but all the current workbook settings, such as passwords and file protection. Does not affect the current workbook. Use this command if you need a temporary copy of the current workbook; for example, to include in an electronic mail message.

Syntax

SAVE.COPY.AS(document_text)

Document_text is the name you want to give the copy of the workbook.

Example

Suppose that you are creating a macro that makes changes to a file called BUDGET95.XLS. Use the following function to save a copy of this file called TEMP.XLS without affecting BUDGET95.XLS:

```
SAVE.COPY.AS("temp.xls")
```

SAVE.DIALOG

Displays the standard Microsoft Excel File Save As dialog box and gets a file name from the user. This function returns the path and file name of the file that has been saved. Use SAVE.AS to automatically save a file with a particular format and other properties.

Syntax

SAVE.DIALOG(init_filename, title, button_text, file_filter, filter_index)

Init_filename Specifies the suggested filename for saving. If omitted, the active workbook's name is used, as returned by the GET.DOCUMENT(1) function.

Title Specifies the default window title on Microsoft Excel for Windows. For Microsoft Excel for the Macintosh, title specifies the prompt string. If omitted, "File Save As" will be used for Microsoft Excel for Windows, and "Save As:" For Microsoft Excel for the Macintosh.

Button_text is the text used for the save button in the dialog. If omitted, "Save" will be used as the default. This argument is ignored on the Microsoft Excel for Windows.

File_filter is the file filtering criteria to use, as text. For Microsoft Excel for Windows, file_filter consists of two parts, a descriptive phrase denoting the file type followed by a comma and then the MS-DOS wildcard file filter specification, as in "Text Files (*.TXT), *.TXT, Add-in Files (*.XLA), *.XLA". Groups of filter specifications are also separated by commas. Each separate pair is listed in the file type drop-down list box. File_filter can include an asterisk (*) to represent any sequence of characters and a question mark (?) to represent any single character. For Microsoft Excel for the Macintosh, file_filter consists of file type codes separated by commas, as in "TEXT,XLA,XLS4". Spaces are significant and should not be inserted before or after the comma separators unless they are part of the file type code.

Filter_index specifies the index number of the default file filtering criteria from 1 to the number of filters specified in file_filter. If omitted or greater than the number of filters present, 1 will be used as the starting index number. The argument is ignored on Microsoft Excel for the Macintosh.

Remarks

- To use multiple MS-DOS wildcard expressions within file_filter for a single file filter type, separate the wildcard expressions with semicolons, as in "VB Files (*.bas; *.txt), *.bas;*.txt".
- If file_filter is omitted, "ALL Files (*.*), *.*" will be used as the default in Microsoft Excel for Windows. The default for Microsoft Excel for the Macintosh is all file types.
- If the user cancels the dialog box, FALSE is returned.

Examples

`SAVE.DIALOG("TRAVEL.XLS","How do you want to save this file?",, "Text Files (*.TXT), *.TXT, Add-in Files (*.XLA), *.XLA, ALL FILES (*.*), *.*")` opens a File Save As dialog box titled "How do you want to save this file?", with "TRAVEL.XLS" as the suggested file name, and with three file filter criteria in the drop-down list box.

Related Function

OPEN.DIALOG   Displays the standard Microsoft Excel File Open dialog box with the specified file filters

SAVE.TOOLBAR

Saves one or more toolbar definitions to a specified file.

Syntax

SAVE.TOOLBAR(bar_id, filename)

Bar_id   is either the name or number of a toolbar whose definition you want to save or an array of toolbar names or numbers whose definitions you want to save. Use an array to save several toolbar definitions at the same time. For detailed information about bar_id, see ADD.TOOL. If bar_id is omitted, all toolbar definitions are saved.

Filename   is text specifying the name of the destination file. If filename does not exist, Microsoft Excel creates a new file. If filename exists, Microsoft Excel overwrites the file. If filename is omitted, Microsoft Excel saves the toolbar or toolbars in Username8.xlb, where "username" is your Windows or network logon name. With Microsoft Windows, Username8.xlb is stored in the directory where Windows is installed; with Apple Macintosh, EXCEL TOOLBARS is stored in the System:Preferences folder

Examples

In Microsoft Excel for Windows, the following macro formula saves Toolbar6 as \EXCDT\TOOLFILE.XLB.

```
SAVE.TOOLBAR("Toolbar6", "\EXCDT\TOOLFILE.XLB")
```

In Microsoft Excel for the Macintosh, the following macro formula saves Toolbar6 as TOOLFILE.

```
SAVE.TOOLBAR("Toolbar6", "TOOLFILE")
```

Related Functions

ADD.TOOL   Adds one or more tools to a toolbar

ADD.TOOLBAR   Creates a new toolbar with the specified tools

OPEN   Opens a workbook

SAVE.WORKBOOK

Equivalent to clicking the Save Workbook command on the File menu in Microsoft Excel version 4.0. Provided for compatibility with Microsoft Excel version 4.0. Saves the workbook to which the active sheet belongs. To save Microsoft Excel version 5.0 or later workbooks, use SAVE.AS.

Syntax

SAVE.WORKBOOK(document_text, type_num, prot_pwd, backup, write_res_pwd, read_only_rec)

SAVE.WORKBOOK?(document_text, type_num, prot_pwd, backup, write_res_pwd, read_only_rec)

For a description of the arguments, see SAVE.AS.

Related Functions

CLOSE   Closes the active window

GET.DOCUMENT   Returns information about a workbook

SAVE   Saves the active workbook

SAVE.AS   Saves a workbook and allows you to specify the name, file type, password, backup file, and location of the workbook

SAVE.WORKSPACE

Equivalent to clicking the Save Workspace command on the File menu. Saves the currently opened workbook or workbooks as a workspace.

Syntax

SAVE.WORKSPACE(name_text)

SAVE.WORKSPACE?(name_text)

Name_text   is the name of the workspace to save.

Related Function

SAVE.AS   Specifies a new filename.

SCALE

Changes the position, formatting, and scaling of axes in a chart. There are five syntax forms of this function.

Syntax 1   Changes the position, formatting, and scaling of the category axis in 2-D charts

Syntax 2   Changes the position, formatting, and scaling of the value axis in 2-D charts

Syntax 3   Changes the position, formatting, and scaling of the category axis in 3-D charts

Syntax 4   Changes the position, formatting, and scaling of the series axis in 3-D charts

Syntax 5   Changes the position, formatting, and scaling of the value axis in 3-D charts

SCALE SYNTAX 1

Equivalent to clicking the Selected Axis command on the Format menu when a chart's category (x) axis is selected, and then clicking the Scale tab. There are five syntax forms of this function. Syntax 1 of SCALE applies if the selected axis is a category (x) axis on a 2-D chart and the chart is not an xy (scatter) chart. Use this syntax of SCALE to change the position, formatting, and scaling of the category axis.

Syntax 1

SCALE(cross, cat_labels, cat_marks, between, max, reverse)

SCALE?(cross, cat_labels, cat_marks, between, max, reverse)

Arguments correspond to text boxes and check boxes in the Scale tab on the Format Axis dialog box. Arguments corresponding to check boxes are logical values. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box.

Cross is a number corresponding to the Value (Y) Axis Crosses At Category number text box. The default is 1. Cross is ignored if max is set to TRUE.

Cat_labels is a number corresponding to the Number Of Categories Between Tick Mark Labels text box. The default is 1.

Cat_marks is a number corresponding to the Number Of Categories Between Tick Marks text box. The default is 1.

Between corresponds to the Value (Y) Axis Crosses Between Categories check box. This argument only applies if cat_labels is set to a number other than 1.

Max corresponds to the Value (Y) Axis Crosses At Maximum Category check box. If max is TRUE, it overrides any setting for cross.

Reverse corresponds to the Categories In Reverse Order check box.

Related Functions

AXES	Controls whether axes on a chart are visible
GRIDLINES	Controls whether chart gridlines are visible
Syntax 2	Changes the position, formatting, and scaling of the value axis in 2-D charts
Syntax 3	Changes the position, formatting, and scaling of the category axis in 3-D charts
Syntax 4	Changes the position, formatting, and scaling of the series axis in 3-D charts
Syntax 5	Changes the position, formatting, and scaling of the value axis in 3-D charts

SCALE SYNTAX 2

Equivalent to clicking the Selected Axes command on the Format menu when a chart's value (y) axis is selected, and then clicking the Scale tab. There are five syntax forms of this function. Syntax 2 of SCALE applies if the selected axis is a value (y) axis on a 2-D chart, or either axis on an xy (scatter) chart. Use this syntax of SCALE to change the position, formatting, and scaling of the value axis.

Syntax 2

SCALE(min_num, max_num, major, minor, cross, logarithmic, reverse, max)

SCALE?(min_num, max_num, major, minor, cross, logarithmic, reverse, max)

The first five arguments correspond to the five range variables on the Scale tab. Each argument can be either the logical value TRUE or a number:

- If an argument is TRUE, Microsoft Excel selects the Auto check box.
- If an argument is a number, that number is used for the variable.

Min_num corresponds to the Minimum check box and is the minimum value for the value axis.

Max_num corresponds to the Maximum check box and is the maximum value for the value axis.

Major corresponds to the Major Unit check box and is the major unit of measure.

Minor corresponds to the Minor Unit check box and is the minor unit of measure.

Cross corresponds to the Category (X) Axis Crosses At text box for the value (y) axis of a 2-D chart or the Value (Y) Axis Crosses At text box for the category (x) axis of an xy (scatter) chart.

The last three arguments are logical values corresponding to check boxes on the Scale tab . If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box.

Logarithmic corresponds to the Logarithmic Scale check box.

Reverse corresponds to the Values In Reverse Order check box.

Max corresponds to the Category (X) Axis Crosses At Maximum Value check box.

Related Functions

Syntax 1 Changes the position, formatting, and scaling of the category axis in 2-D charts

Syntax 3 Changes the position, formatting, and scaling of the category axis in 3-D charts

Syntax 4 Changes the position, formatting, and scaling of the series axis in 3-D charts

Syntax 5 Changes the position, formatting, and scaling of the value axis in 3-D charts

SCALE SYNTAX 3

Equivalent to clicking the Selected Axes command on the Format menu when a chart's category (x) axis is selected, and then click the Scale tab. There are five syntax forms of this function. Syntax 3 of SCALE applies if the selected axis is a category (x) axis on a 3-D chart. Use this syntax of SCALE to change the position, formatting, and scaling of the category axis.

Syntax 3

SCALE(cat_labels, cat_marks, reverse, between)

SCALE?(cat_labels, cat_marks, reverse, between)

Cat_labels is a number corresponding to the Number Of Categories Between Tick-Mark Labels box. The default is 1. Cat_labels can also be a logical value. If TRUE, an automatic setting will be used. If FALSE, or omitted, the number will be used.

Cat_marks is a number corresponding to the Number Of Categories Between Tick Marks text box. The default is 1. Cat_marks can also be a logical value. If TRUE, an automatic setting will be used. If FALSE, or omitted, the number will be used.

Reverse corresponds to the Categories In Reverse Order check box. If reverse is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box.

Between corresponds to the Value (Z) Axis Crosses Between Categories check box. If between is TRUE, Microsoft Excel selects the check box and the data points appear between categories. If between is FALSE or omitted, Microsoft Excel clears the check box.

Related Functions

Syntax 1 Changes the position, formatting, and scaling of the category axis in 2-D charts

Syntax 2 Changes the position, formatting, and scaling of the value axis in 2-D charts

Syntax 4 Changes the position, formatting, and scaling of the series axis in 3-D charts

Syntax 5 Changes the position, formatting, and scaling of the value axis in 3-D charts

SCALE SYNTAX 4

Equivalent to clicking the Selected Axes command on the Format menu when a chart's value (y) axis is selected, and then clicking the Scale tab. There are five syntax forms of this function. Syntax 4 of SCALE applies if the selected axis is a series (y) axis on a 3-D chart. Use this syntax of SCALE to change the position, formatting, and scaling of the series axis.

Syntax 4

Series (y) axis, 3-D chart

SCALE(series_labels, series_marks, reverse)

SCALE?(series_labels, series_marks, reverse)

Series_labels is a number corresponding to the Number Of Series Between Tick Labels text box. The default is 1. Series_labels can also be a logical value. If TRUE, and automatic setting will be used. If FALSE, or omitted, the number will be used.

Series_marks is a number corresponding to the Number Of Series Between Tick Marks text box. The default is 1. Series_marks can also be a logical value. If TRUE, and automatic setting will be used. If FALSE, or omitted, the number will be used.

Reverse is a logical value that corresponds to the Series In Reverse Order check box on the Scale tab. If reverse is TRUE, Microsoft Excel displays the series in reverse order; if FALSE or omitted, Microsoft Excel displays the series normally.

Related Functions

Syntax 1 Changes the position, formatting, and scaling of the category axis in 2-D charts

Syntax 2 Changes the position, formatting, and scaling of the value axis in 2-D charts

Syntax 3 Changes the position, formatting, and scaling of the category axis in 3-D charts

Syntax 5 Changes the position, formatting, and scaling of the value axis in 3-D charts

SCALE SYNTAX 5

Equivalent to clicking the Selected Axes command on the Format menu when a chart's value (z) axis is selected, and then clicking the Scale tab. There are five syntax forms of this function. Syntax 5 of SCALE applies if the selected axis is a value (z) axis on a 3-D chart. Use this syntax of SCALE to change the position, formatting, and scaling of the value axis.

Syntax 5

SCALE(min_num, max_num, major, minor, cross, logarithmic, reverse, min)

SCALE?(min_num, max_num, major, minor, cross, logarithmic, reverse, min)

The first five arguments correspond to the five range variables in the Format Axis dialog box, as shown in the following list. Each argument can be either the logical value TRUE or a number.

- If TRUE or omitted, the Auto check box is selected.
- If a number, that number is used.

Min_num corresponds to the Minimum check box and is the minimum value for the value axis.

Max_num corresponds to the Maximum check box and is the maximum value for the value axis.

Major corresponds to the Major Unit check box and is the major unit of measure.

Minor corresponds to the Minor Unit check box and is the minor unit of measure.

Cross corresponds to the Floor (XY Plane) Crosses At check box.

The last three arguments are logical values corresponding to check boxes on the Scale tab. If an argument is TRUE, Microsoft Excel selects the check box; if FALSE, Microsoft Excel clears the check box.

Logarithmic corresponds to the Logarithmic Scale check box.

Reverse corresponds to the Values In Reverse Order check box.

Min corresponds to the Floor (XY Plane) Crosses At Minimum Value check box.

Related Functions

Syntax 1	Changes the position, formatting, and scaling of the category axis in 2-D charts
Syntax 2	Changes the position, formatting, and scaling of the value axis in 2-D charts
Syntax 3	Changes the position, formatting, and scaling of the category axis in 3-D charts
Syntax 4	Changes the position, formatting, and scaling of the series axis in 3-D charts

SCENARIO.ADD

Equivalent to clicking the Scenarios command on the Tools menu and then clicking the Add button. Defines the specified values as a scenario. A scenario is a set of values to be used as input for a model on your worksheet.

Syntax

SCENARIO.ADD(**scen_name**, value_array, changing_ref, scen_comment, locked, hidden)

Scen_name is the name of the scenario you want to define.

Value_array is a horizontal array of values you want to use as input for the model on your worksheet.

- Any entry that would be valid for a cell in your model can be a value in value_array.
- The values must be arranged in the same order as the model's changing cells. The changing cells are listed in the Changing Cells box in the Scenario Manager dialog box.
- If value_array is omitted, it is assumed to contain the current values of the changing cells.

Changing_ref is a reference to cells you want to define as changing cells for a scenario.

- If omitted, uses the changing cells for the last scenario defined for the sheet.

- If changing_ref contains nonadjacent references, you must separate the reference areas by commas (or other list separator). If you are using A1-style references, then you must enclose reference in an extra set of parentheses.

Scen_comment is text specifying a descriptive comment for the scenario defined by scen_name.

Locked is a logical value that corresponds to the Prevent Changes check box in the Add or Edit Scenario dialogs boxes. If TRUE or omitted , prevents users from changing values in a scenario. If FALSE, users are allowed to make changes to the scenario. The locking will not become enabled until the sheet is protected with the Protect Sheet command from the Protection submenu on the Tools menu.

Hidden is a logical value that corresponds to the Hide check box in the Add or Edit Scenario dialog boxes. If TRUE, the scenario will be hidden from view from the users and will not appear in the Scenario Manager dialog box. If FALSE or omitted, the scenario will remain unhidden. The scenario will not become hidden until the sheet is protected with the Protect Sheet command from the Protection submenu on the Tools menu.

Related Functions

REPORT.DEFINE Creates a report

SCENARIO.GET Returns the specified information about the scenarios defined on your worksheet

SCENARIO.CELLS

Equivalent to clicking the Scenarios command on the Tools menu and then editing the Changing Cells box. Defines the changing cells for a model on your worksheet. Changing cells are the cells into which values will be entered when you display a scenario. If you have only one set of changing cells on your sheet, SCENARIO.CELLS will change the changing cells for all scenarios. If your sheet has scenarios defined with multiple sets of changing cell, this function returns an error and the macro is halted. This function is provided for compatibility with Microsoft Excel version 4.0. Use SCENARIO.EDIT with the changing_ref argument instead of SCENARIO.CELLS if you want to change the changing cells of a scenario.

Syntax

SCENARIO.CELLS(**changing_ref**)

SCENARIO.CELLS? (changing_ref)

Changing_ref is a reference to the cells you want to define as changing cells for the model. If changing_ref contains nonadjacent references, you must separate the reference areas by commas and enclose changing_ref in an extra set of parentheses.

Related Function

SCENARIO.EDIT Equivalent to clicking the Scenarios command on the Tools menu and then clicking the Edit button

SCENARIO.DELETE

Equivalent to clicking the Scenarios command on the Tools menu, clicking a scenario, and then clicking the Delete button. Deletes the specified scenario.

Syntax

SCENARIO.DELETE(**scen_name**)

Scen_name   is the name of the scenario you want to delete.

Related Functions

SCENARIO.GET   Returns the specified information about the scenarios defined on your worksheet

SCENARIO.ADD   Equivalent to clicking the Scenario Manager command on the Tools menu and then clicking the Add button

SCENARIO.EDIT   Equivalent to clicking the Scenario Manager command on the Tools menu and then clicking the Edit button

SCENARIO.EDIT

Equivalent to clicking the Scenarios command from the Tools menu and then clicking the Edit button.

Syntax

SCENARIO.EDIT(**scen_name**, new_scenname, value_array, changing_ref, scen_comment, locked, hidden)

SCENARIO.EDIT?(scen_name, new_scenname, value_array, changing_ref, scen_comment, locked, hidden)

Scen_name   is the name of the scenario that you want to edit.

New_scenname   is the new name you want to give to the scenario.

Value_array   is a horizontal array of values that you want to use for the scenario.

- If value_array is omitted but changing_ref is specified, Scenario Manager uses the values in changing_ref as value_array.
- Value_array must match the dimensions of changing_ref for the scenario being edit.

Changing_ref   is a reference to cells you want to define as changing cells for a scenario.

Scen_comment   is text specifying a descriptive comment for the scenario you want to edit.

Locked   is a logical value that corresponds to the Prevent Changes check box in the Add or Edit Scenario dialogs boxes. If TRUE or omitted , prevents users from changing values in a scenario. If FALSE, users are allowed to make changes to the scenario. The locking will not become enabled until the sheet is protected with the Protect Sheet command from the Protection submenu on the Tools menu.

Hidden   is a logical value that corresponds to the Hide check box in the Add or Edit Scenario dialog boxes. If TRUE, the scenario will be hidden from view from the users. If FALSE or omitted, the scenario will remain unhidden. The scenario will not become hidden until the sheet is hidden with the Hide command from the Window menu.

Related Functions

SCENARIO.GET   Returns the specified information about the scenarios defined on your worksheet

SCENARIO.ADD   Equivalent to clicking the Scenario Manager command on the Tools menu and then clicking the Add button

SCENARIO.DELETE   Equivalent to clicking the Scenario Manager command on the Tools menu and then selecting a scenario and clicking the Delete button

SCENARIO.GET

Returns the specified information about the scenarios defined on your worksheet.

Syntax

SCENARIO.GET(**type_num**, scen_name)

Type_num   is a number from 1 to 8 specifying the type of information you want.

Type_num	**Information returned**
1	A horizontal array of all scenario names in the form of text
2	A reference to the set of changing cells of scen_name (specified in the Changing Cells box of the Scenario Manager dialog box). If scen_name is omitted, the first scenario is used.
3	A reference to the result cells (specified in the Result Cells box in the Scenario Summary dialog box)
4	An array of scenario values for the scenario scen_name . Each scenario is in a separate row. If scen_name is omitted, the first scenario is used.
5	Comment, as text, for the scenario
6	Returns TRUE if the specified scenario is locked to prevent changes; FALSE, if unlocked. Scen_name is required.
7	Returns TRUE if the specified scenario is hidden; FALSE, if visible to the user. Scen_name is required.
8	Returns the user name of the person who last modified the scenario by either adding or editing a scenario. Scen_name is required.

Scen_name   is the name of the scenario that you want information about. Ignored if type_num equals 1 or 3.

Remarks

In the returned array of scenario values, the number of rows is the number of scenarios, and the number of columns is the number of changing cells.

SCENARIO.MERGE

Equivalent to choosing the Scenarios command from the Tools menu and then selecting Merge. This function merges scenarios from other sheets onto the active sheet. A scenario is a set of values to be used as input for a model on your worksheet.

Syntax

SCENARIO.MERGE(source_file)

SCENARIO.MERGE?(source_file)

Source_file is the name of the book and sheet from which you want to merge scenarios onto the active sheet.

Related Function

SCENARIO.GET Returns the specified information about the scenarios defined on your worksheet

SCENARIO.SHOW

Equivalent to clicking the Scenarios command on the Tools menu and then selecting a scenario and clicking the Show button. Recalculates a model using the specified scenario and displays the result.

Syntax

SCENARIO.SHOW(**scen_name**)

Scen_name is the name of the previously defined scenario whose values you want to switch to.

SCENARIO.SHOW.NEXT

Equivalent to clicking the Scenarios command on the Tools menu, selecting the next scenario from the Scenarios list, and clicking the Show button. Recalculates a model using the next scenario and displays the result.

Syntax

SCENARIO.SHOW.NEXT( )

Remarks

After displaying the last scenario, running SCENARIO.SHOW.NEXT again displays the first scenario.

SCENARIO.SUMMARY

Equivalent to clicking the Scenarios command on the Tools menu and then clicking the Summary button. Generates a table summarizing the results of all the scenarios for the model on your worksheet.

Syntax

SCENARIO.SUMMARY(result_ref, report_type)

SCENARIO.SUMMARY?(result_ref, report_type)

Result_ref is a reference to the result cells you want to include in the summary report. Normally, result_ref refers to one or more cells containing the formulas that depend on the changing cell values for your model—that is, the cells that show the results of a particular scenario.

- If result_ref is omitted, no result cells are included in the report.
- If result_ref contains nonadjacent references, you must separate the reference areas by commas and enclose result_ref in an extra set of parentheses.

Report_type is a number specifying the type of report desired.

Report_type	**Type of Report**
1 or omitted	A scenario summary report (Microsoft Excel version 4.0)
2	A scenario PivotTable report. Requires result_ref.

Remarks

- SCENARIO.SUMMARY generates a summary table of the changing cell and result cell values for each scenario.
- The table is generated on a new sheet in the current workbook. The sheet becomes active after SCENARIO.SUMMARY runs.

SCROLLBAR.PROPERTIES

Sets the properties of the scroll bar and spinner button on a worksheet or dialog sheet.

Syntax

SCROLLBAR.PROPERTIES(value, min, max, inc, page, link, 3d_shading)

SCROLLBAR.PROPERTIES?(value, min, max, inc, page, link, 3d_shading)

Value is the value of the control, and can range from min to max, inclusive. It designates where the scroll bar button is positioned along the scroll bar.

Min is a number specifying the minimum value that the scroll bar can have. This number ranges from 0 to 30,000, but cannot be greater than the maximum value given in max.

Max is a number specifying the maximum value that the scroll bar can have. This number ranges from 0 to 30,000.

Inc is a number specifying the increment that the value is adjusted by when the scrollbar arrow is clicked.

Page is a number specifying the increment that the value is adjusted by when the page scroll region of a scroll bar is clicked.

Link is the cell on the macro sheet to which the scroll bar value is linked. Whenever the scroll bar control is changed, the value of the control is entered into the cell. Similarly, whenever the value in the cell is changed, the setting for the scroll bar is also changed. To clear the link, set this value to an empty string.

3d_shading is a logical value that specifies whether the scroll bar or spinner button appears as 3-D. If TRUE, the scroll bar or spinner button will appear as 3-D. If FALSE or omitted, the scroll bar or spinner button will not be 3-D. The argument is available for only worksheets.

Related Functions

PUSHBUTTON.PROPERTIES	Sets the properties of the push button control
EDITBOX.PROPERTIES	Sets the properties of an edit box on a worksheet or dialog sheet

SELECT

Equivalent to selecting cells or changing the active cell. There are three syntax forms of SELECT. Use syntax 1 to select a cell on a worksheet or macro sheet; use one of the other syntax forms to select worksheet or macro sheet objects or chart items.

Syntax 1	Selects cells
Syntax 2	Selects objects on worksheets
Syntax 3	Selects chart objects

SELECT SYNTAX 1

Equivalent to selecting cells or changing the active cell. There are three syntax forms of SELECT. Use syntax 1 to select a cell on a worksheet or macro sheet; use one of the other syntax forms to select worksheet or macro sheet objects or chart items.

Syntax

SELECT(selection, active_cell)

Selection is the cell or range of cells you want to select. Selection can be a reference to the active worksheet, such as !$A$1:$A$3 or !Sales, or an R1C1-style reference to a cell or range relative to the active cell in the current selection, such as "R[-1]C[-1]:R[1]C[1]". The reference must be in text form. If selection is omitted, the current selection is used.

Active_cell is the cell in selection you want to make the active cell. Active_cell can be a reference to a single cell on the active worksheet, such as !$A$1, or an R1C1-style reference relative to the active cell, such as "R[-1]C[-1]". The reference must be in text form. If active_cell is omitted, SELECT makes the cell in the upper-left corner of selection the active cell.

Remarks

- Active_cell must be within selection. If it is not, an error message is displayed and SELECT returns the #VALUE! error value.

- If you are recording a macro using relative references, Microsoft Excel records the action using R1C1-style relative references in the form of text.
- If you are recording using absolute references, Microsoft Excel records the action using R1C1-style absolute references in the form of text.
- You cannot give an external reference to a specific sheet as the selection argument. The sheet on which you want to make a selection must be active when you use SELECT. Use FORMULA.GOTO to make a selection on another sheet in the same workbook or in another workbook.

Tip You can enter data in a cell without selecting the cell by using the reference arguments to the CUT, COPY, or FORMULA functions.

Examples

The following macro formula selects cells C3:E5 on the active worksheet and makes C5 the active cell:

```
SELECT(!$C$3:$E$5, !$C$5)
```

If the active cell is C3, the following macro formula selects cells E5:G7 and makes cell F6 the active cell in the selection:

```
SELECT("R[2]C[2]:R[4]C[4]", "R[1]C[1]")
```

You can also make multiple nonadjacent selections with SELECT. The following macro formula selects a number of nonadjacent ranges:

```
SELECT("R1C1, R3C2:R4C3, R8C4:R10C5")
```

The following sequence of macro formulas moves the active cell right, left, down, and up within the selection, just as TAB, SHIFT+TAB, ENTER, and SHIFT+ENTER do:

```
SELECT(, "RC[1]")

SELECT(, "RC[-1]")

SELECT(, "R[1]C")

SELECT(, "R[-1]C")
```

Use SELECT with the OFFSET function to select a new range a specified distance away from the current range. For example, the following macro formula selects a range that is the same size as the current range, one column over:

```
SELECT(OFFSET(SELECTION(), 0, 1))
```

Related Functions

ACTIVE.CELL Returns the reference of the active cell

SELECT.SPECIAL   Selects a group of cells belonging to a category

SELECTION   Returns the reference of the selection

SELECT Syntax 2   Selects objects on worksheets

SELECT Syntax 3   Selects chart objects

SELECT SYNTAX 2

Equivalent to selecting objects on a chart, worksheet, or macro sheet. There are three syntax forms of SELECT. Use syntax 2 to select an object on which to perform an action; use one of the other syntax forms to select cells on a worksheet or macro sheet or items on a chart.

Syntax

SELECT(object_id_text, replace)

Object_id_text   is text that identifies the object to select. Object_id_text can be the name of more than one object. To give the name of more than one object, use the following format:

```
SELECT("Oval 3, Arc 2, Line 4")
```

The last item in the object_id_text list will be the active object. The active object is important when moving and sizing a group of objects. A multiple selection of objects is moved and sized relative to the upper-left corner of the active object.

Replace   is a logical value that specifies whether previously selected objects are included in the selection. If replace is TRUE or omitted, Microsoft Excel only selects the objects specified by object_id_text; if FALSE, it includes any objects that were previously selected. For example, if a button is selected and a SELECT formula selects an arc and an oval, TRUE leaves only the arc and oval selected, and FALSE includes the button with the arc and oval.

Remarks

Objects can be identified by their object type and number as described in CREATE.OBJECT, or by the unique number that specifies the order of their creation. For example, if the third object you create is an oval, you could use either "oval 3" or "3" as object_id_text.

Examples

The following macro formulas each select a number of objects and specify Arc 2 as the active object:

```
SELECT("Oval 3, Arc 1, Line 4, Arc 2")

SELECT("3, 1, 4, 2")
```

Related Functions

FORMAT.MOVE   Moves the selected object

FORMAT.SIZE   Changes the size of the selected objects

GET.OBJECT   Returns information about an object

SELECTION   Returns the reference of the selection

SELECT Syntax 1	Selects cells
SELECT Syntax 3	Selects chart objects

SELECT SYNTAX 3

Selects a chart object as specified by the selection code item_text. There are three syntax forms of SELECT. Use syntax 3 to select a chart item to which you want to apply formatting; use one of the other syntax forms to select cells or objects on a worksheet or macro sheet.

Syntax

SELECT(item_text, single_point)

Item_text is a selection code from the following table which specifies which chart object to select.

To select	Item_text
Entire chart	"Chart"
Plot area	"Plot"
Legend	"Legend"
Primary chart value axis	"Axis 1"
Primary chart category axis	"Axis 2"
Secondary chart value axis or 3-D series axis	"Axis 3"
Secondary chart category axis	"Axis 4"
Chart title	"Title"
Label for the primary chart value axis	"Text Axis 1"
Label for the primary chart category axis	"Text Axis 2"
Label for the primary chart series axis	"Text Axis 3"
nth floating text item	"Text n"
nth arrow	"Arrow n"
Major gridlines of value axis	"Gridline 1"

Minor gridlines of value axis	"Gridline 2"
Major gridlines of category axis	"Gridline 3"
Minor gridlines of category axis	"Gridline 4"
Major gridlines of series axis	"Gridline 5"
Minor gridlines of series axis	"Gridline 6"
Primary chart droplines	"Dropline 1"
Secondary chart droplines	"Dropline 2"
Primary chart hi-lo lines	"Hiloline 1"
Secondary chart hi-lo lines	"Hiloline 2"
Primary chart up bar	"UpBar1"
Secondary chart up bar	"UpBar2"
Primary chart down bar	"DownBar1"
Secondary chart down bar	"DownBar2"
Primary chart series line	"Seriesline1"
Secondary chart series line	"Seriesline2"
Entire series	"Sn"
Data associated with point m in series n if single_point is TRUE	"SnPm"
Text attached to point m of series n	"Text SnPm"
Series title text of series n of an area chart	"Text Sn"
Base of a 3-D chart	"Floor"
Back of a 3-D chart	"Walls"

Corners of a 3-D chart	"Corners"
Trend line	"SnTm"
Error bars	"SnEm"
Legend Marker	"Legend Marker n"
Legend Entry	"Legend Entry n"

For trend lines and error bars, the value m can be X or Y, depending on which point you want to select. If m is blank, selects both.

Single_point is a logical value that determines whether to select a single point. Single_point is available only when item_text is "SnPm".

- If single_point is TRUE, Microsoft Excel selects a single point.
- If single_point is FALSE or omitted, Microsoft Excel selects a single point if there is only one series in the chart or selects the entire series if there is more than one series in the chart.
- If you specify single_point when item_text is any value other than "SnPm", SELECT returns an error value.

Examples

`SELECT("Chart")` selects the entire chart.

`SELECT("Dropline 2")` selects the droplines of an overlay chart.

`SELECT("S1P3", TRUE)` selects the third point in the first series.

`SELECT("Text S1")` selects the series title text of the first series in an area chart.

Related Functions

SELECTION	Returns the reference of the selection
SELECT Syntax 1	Selects cells
SELECT Syntax 2	Selects objects on worksheets

SELECT.ALL

Equivalent to selecting all the sheets in a workbook.

Syntax

SELECT.ALL( )

SELECT.CHART

Equivalent to the Select Chart command on the Chart menu in Microsoft Excel version 4.0. This function is equivalent to using the third form of SELECT with "Chart" as the item_text argument.

Syntax

SELECT.CHART( )

Remarks

This function is included for compatibility with macros written with Microsoft Excel for the Macintosh version 1.5 or earlier.

Related Function

SELECT   Selects a chart object

SELECT.END

Selects the cell at the edge of the range or the first cell of the next range in the direction specified. Equivalent to pressing CTRL+ARROW in Microsoft Excel for Windows or COMMAND+ARROW in Microsoft Excel for the Macintosh.

Syntax

SELECT.END(**direction_num**)

Direction_num   is a number from 1 to 4 indicating the direction in which to move.

Direction_num	Direction
1	Left (equivalent to CTRL+LEFT ARROW or COMMAND+LEFT ARROW)
2	Right (equivalent to CTRL+RIGHT ARROW or COMMAND+RIGHT ARROW)
3	Up (equivalent to CTRL+UP ARROW or COMMAND+UP ARROW)
4	Down (equivalent to CTRL+DOWN ARROW or COMMAND+DOWN ARROW)

Related Function

SELECT.LAST.CELL   Selects the last cell on a worksheet or macro sheet that contains a formula, value, or format or that is referred to in a formula or name

SELECTION

Returns the reference or object identifier of the selection as an external reference. Use SELECTION to return information about the current selection for use in other macro formulas.

Syntax

SELECTION( )

If a cell or range of cells is selected, Microsoft Excel returns the corresponding external reference. If an object is selected, Microsoft Excel returns the object identifier listed in the following table.

Item selected	Identifier returned
Imported graphic	Picture n
Linked graphic	Picture n
Chart picture	Picture n
Linked chart	Chart n
Range	Picture n
Linked range	Picture n
Text box	Text n
Button	Button n
Rectangle	Rectangle n
Oval	Oval n
Line	Line n
Arc	Arc n
Group	Group n
Freehand drawing or polygon	Drawing n

SELECTION also returns the identifiers of chart items. The identifiers returned are the same as the identifiers you specify when you use the SELECT function. For a list of these identifiers, see the description of item_text in SELECT.

If you select cells and use the value returned by SELECTION in a function or operation, you usually get the value contained in the selection instead of its reference. References are automatically converted to the contents of the reference. If you want to work with the actual reference, use SET.NAME to assign a name to it, even if the reference refers to objects. See the last example following. You can also use the REFTEXT function to convert the reference to text, which you can then store or manipulate.

Remarks

- If an object is selected, SELECTION returns the identifier of the object. If multiple objects are selected, it returns the identifiers of all the selected objects, as a string separated by commas.
- If more than 1024 characters would be returned, SELECTION returns the #VALUE! error value.

Examples

If the sheet in the active window is named SHEET1 in the workbook BOOK1, and if A1:A3 is the selection, then:

```
SELECTION() equals [BOOK1]SHEET1!A1:A3
```

The following macro formula moves the current selection one row down:

```
SELECT(OFFSET(SELECTION(), 1, 0))
```

The above formula is particularly useful for moving incrementally through a database to add or modify records.

The following macro formula defines the name "EntryRange" on the active sheet to refer to one row below the current selection on the active sheet:

```
DEFINE.NAME("EntryRange", OFFSET(SELECTION(), 1, 0))
```

The following macro formula defines the name "Objects" on your macro sheet to refer to the object names in the current multiple selection:

```
SET.NAME("Objects", SELECTION())
```

Related Functions

ACTIVE.CELL	Returns the reference of the active cell
SELECT	Selects a cell, graphic object, or chart

SELECT.LAST.CELL

Equivalent to choosing the Special button from the Go To dialog box and selecting the Last Cell option. The Go To dialog box appears when you choose the Go To command from the Edit menu. Selects the cell at the intersection of the last row and column that contains a formula, value, or format, or that is referred to in a formula or name.

Syntax

SELECT.LAST.CELL( )

Related Function

SELECT.END   Selects the last cell in a range

SELECT.LIST.ITEM

Selects an item in a list box or in a group box.

Syntax

SELECT.LIST.ITEM(**index_num**, selected_logical)

Index_num   is the index number of the item to select. Using zero will deselect all items. Adding 1 to the number of items in the list will select all the items specified.

Selected_logical   is a number that specifies the selection mode of the list box. Zero is single selection. 1 is simple multi-select. 2 is extended multi-select.

Related Functions

ADD.LIST.ITEM   Adds an item in a list box or drop-down control on a worksheet or dialog sheet control

REMOVE.LIST.ITEM   Removes an item in a list box or drop-down box

SELECT.PLOT.AREA

Equivalent to clicking the Select Plot Area command on the Chart menu in Microsoft Excel version 4.0. Selects the plot area of the active chart.

Syntax

SELECT.PLOT.AREA( )

Remarks

SELECT.PLOT.AREA is included only for compatibility with previous versions of Microsoft Excel for the Macintosh. SELECT.PLOT.AREA is the same as the SELECT("Plot") function.

Related Function

SELECT   Selects a cell, graphic object, or chart

SELECT.SPECIAL

Equivalent to clicking the Go To command on the Edit menu and then selecting the Special button. Use SELECT.SPECIAL to select groups of similar cells in one of a variety of categories.

Syntax

SELECT.SPECIAL(**type_num**, value_type, levels)

SELECT.SPECIAL?(type_num, value_type, levels)

Type_num   is a number from 1 to 13 corresponding to options in the Go To Special dialog box and describes what to select.

Type_num	Description
1	Notes/comments
2	Constants
3	Formulas
4	Blanks
5	Current region
6	Current array
7	Row differences
8	Column differences
9	Precedents
10	Dependents
11	Last cell
12	Visible cells only (outlining)
13	All objects

Value_type is a number specifying which types of constants or formulas you want to select. Value_type is available only when type_num is 2 or 3.

Value_type	Selects
1	Numbers
2	Text
4	Logical values
16	Error values

These values can be added to select more than one type. The default for value_type is 23, which select all value types.

Levels is a number specifying how precedents and dependents are selected. Levels is available only when type_num is 9 or 10. The default is 1.

Levels	Selects
1	Direct only
2	All levels

SEND.KEYS

Sends keystrokes to the active application just as if they were typed at the keyboard. Use SEND.KEYS to send keystrokes that perform actions and execute commands to applications you are running with Microsoft Excel's other dynamic data exchange (DDE) functions.

Syntax

SEND.KEYS(**key_text**, wait_logical)

Note This function is available only in Microsoft Excel for Windows.

Key_text is the key or key combination you want to send to another application. The format for key_text is described in the ON.KEY function.

Wait_logical is a logical value that determines whether the macro continues before the actions caused by key_text are carried out.

- If wait_logical is TRUE, Microsoft Excel waits for the keys to be processed before returning control to the macro.
- If wait_logical is FALSE or omitted, the macro continues running without waiting for the keys to be processed.

Remarks

If Microsoft Excel is the active application, wait_logical is assumed to be FALSE, even if you enter wait_logical as TRUE. This is because if wait_logical is TRUE, Microsoft Excel waits for the keys to be processed in the other application before returning control to the macro. Microsoft Excel doesn't process keys while a macro is running.

Example

The following macro uses the Calculator application in Microsoft Excel for Windows to multiply some numbers, and then cuts the result and pastes it into Microsoft Excel.

```
=EXEC("CALC.EXE", 1)

=SEND.KEYS("10*30", TRUE)

=SEND.KEYS("~", TRUE)
```

```
=SEND.KEYS("%ec", TRUE)

=APP.ACTIVATE(, FALSE)

=SELECT(!B1)

=PASTE()

=RETURN()
```

Related Functions

APP.ACTIVATE   Switches to an application

EXECUTE   Carries out a command in another application

ON.KEY   Runs a macro when a specified key is pressed

SEND.MAIL

Equivalent to clicking the Send Mail command on the File menu. Sends the active workbook using email.

Syntax

SEND.MAIL(**recipients**, subject, return_receipt)

SEND.MAIL?(recipients, subject, return_receipt)

Important   To use SEND.MAIL in Microsoft Excel for Windows, you must be using a mail client that supports the Messaging Applications Programming Interface (MAPI) or Vendor-Independent Messaging (VIM). To use SEND.MAIL in Microsoft Excel for the Macintosh, you must be using Microsoft Mail version 2.0 or later.

Recipients   is the name of the person to whom you want to send the mail. The name should be given as text.

- To specify more than one name, give the list of names as an array. For example, SEND.MAIL({"John", "Paul", "George", "Ringo"}) would send the active workbook to the four names in the array. You can also refer to a range on a sheet or macro sheet that contains a list of names to whom you want the mail to be sent.
- To send mail to users on different Microsoft Mail for the Macintosh servers, specify the server name along with the user name. The following text, as the recipients argument, sends mail to wandagr on server2, gregpr on the current server, and victorge on server7:

  {"wandagr@server2", "gregpr", "victorge@server7"}

Subject   is a text string that specifies the subject of the message. If subject is omitted, the name of the active workbook is used as the subject.

Return_receipt   is a logical value that corresponds to the Return Receipt check box. If return_receipt is TRUE, Microsoft Excel selects the check box and sends a return receipt; if FALSE or omitted, Microsoft Excel clears the check box.

Related Function

OPEN.MAIL Opens files sent via Microsoft Mail that Microsoft Excel can open

SEND.TO.BACK

Sends the selected object or objects to the back. Use SEND.TO.BACK to position selected objects behind other objects.

If the selection is not an object or a group of objects, SEND.TO.BACK returns the #VALUE! error value and interrupts the macro.

Syntax

SEND.TO.BACK( )

Related Function

BRING.TO.FRONT Brings selected objects to the front

SERIES

Charts Only

Represents a data series in the active chart. SERIES is used only in charts; you cannot enter it on a sheet or macro sheet. You normally create or change data series by using the Chart Wizard or EDIT.SERIES macro function, which is equivalent to the Edit Series command on the Chart menu in Microsoft Excel version 4.0. However, you can edit a data series manually by selecting it, switching to the formula bar, and typing the changes.

Syntax

SERIES(name_ref, categories, **values, plot_order**)

Name_ref is the name of the data series. It can be an external reference to a single cell or a name defined as a single cell. Name_ref can also be text enclosed in quotation marks (for example, "Projected Sales").

Categories is an external reference to the name of the workbook and to the cells that contain one of the following sets of data:

- Category labels for all charts except xy (scatter) charts
- X-coordinate data for xy (scatter) charts

Values is an external reference to the name of the workbook and to the cells that contain values (or y-coordinate data in scatter charts).

Plot_order is an integer specifying whether the series is plotted first, second, or third, and so on, in the chart. No two series can have the same plot_order.

Remarks

- Categories and values can be arrays or references to a multiple selection, although they cannot be names that refer to a multiple selection. If you specify a multiple selection for any of these arguments, make sure you include the necessary sets of parentheses so that Microsoft Excel does not treat the components of the references as separate arguments.

- If either categories or values is a multiple selection, then all areas in that selection must be either vertical (more rows than columns) or horizontal (more columns than rows).

Related Functions

CHART.WIZARD Creates and formats a chart

EDIT.SERIES Creates or changes a chart series

SERIES.AXES

Equivalent to the Axis Tab in the Format Data Series dialog box. Changes the axis on which a series is plotted. This function is for compatibility with Microsoft Excel versions earlier than Microsoft Excel 97.

Syntax

SERIES.AXES(axis)

Axis is a number specifying on which axis to plot the data series: use 1 for primary axis, 2 for secondary axis.

SERIES.ORDER

Changes the order of series in a chart.

Syntax

SERIES.ORDER(chart_num, old_series_num, new_series_num)

Chart_num is the number of the group containing the series you want to change

Old_series_num is the current number of the series in the group.

New_series_num is the new number you want for the series in the group.

SERIES.X

Equivalent to the X Values tab in the Format Data Series dialog box. Specifies the category labels (x values) for a data series. This function is for compatibility with Microsoft Excel versions earlier than Microsoft Excel 97.

Syntax

SERIES.X(x_ref)

X-ref is an external reference in the form of text specifying the range containing the category labels (or x values for a scatter (xy) chart) you want to use.

Related Function

SERIES.Y Specifies the name and values for a data series

SERIES.Y

Equivalent to the Name and Values tab in the Format Data Series dialog box. Specifies the name and values for a data series. This function is for compatibility with Microsoft Excel versions earlier than Microsoft Excel 97.

Syntax

SERIES.Y(name_ref, y_ref)

Name_ref is text or an external reference in the form of text specifying the name for the data series that appears in the legend for the chart.

Y_ref is an external reference in the form of text specifying the range containing the values for the data series.

Related Function

SERIES.X Specifies the category labels (x values) for a data series

SET.CONTROL.VALUE

Changes the value for the active control, such as a list box, drop-down box, check box, option button, scroll bar, and spinner button.

Syntax

SET.CONTROL.VALUE(value)

Value is the value you want to change. The control interprets this value as follows:

Control	**Value is**
List box	The index of the selected item. If zero, then no item is selected.
Drop-down box	The index of the selected item. If zero, then no item is selected.
Check box	0 = Off 1 = On 2 = Mixed
Option button	0= Off 1 = On
Scroll bar	The numeric value of the control, between the maximum and minimum values
Spinner button	The numeric value of the control, between the maximum and minimum values

Related Functions

ADD.LIST.ITEM   Adds an item in a list box or drop-down control on a worksheet or dialog sheet control

REMOVE.LIST.ITEM   Removes an item in a list box or drop-down box

SELECT.LIST.ITEM   Selects an item in a list box or in a group box

CHECKBOX.PROPERTIES   Sets various properties of check box and option box controls

SCROLLBAR.PROPERTIES   Sets the properties of the scroll bar and spinner controls

SET.CRITERIA

Equivalent to clicking the Set Criteria command on the Data menu in Microsoft Excel version 4.0. Defines the name Criteria for the selected range on a sheet or macro sheet.

Syntax

SET.CRITERIA( )

Related Functions

SET.DATABASE   Equivalent to clicking the Set Database command on the Data menu in Microsoft Excel version 4.0

SET.EXTRACT   Equivalent to clicking the Set Extract command on the Data menu in Microsoft Excel version 4.0

SET.DATABASE

Equivalent to clicking the Set Database command on the Data menu in Microsoft Excel version 4.0. Defines the name Database for the selected range on a sheet or macro sheet.

Syntax

SET.DATABASE( )

Related Functions

SET.CRITERIA   Equivalent to clicking the Set Criteria command on the Data menu in Microsoft Excel version 4.0

SET.EXTRACT   Equivalent to clicking the Set Extract command on the Data menu in Microsoft Excel version 4.0

SET.DIALOG.DEFAULT

Sets which button is automatically pressed (the default button) when the user presses ENTER. While running, this default button is visually recognized by its thick border. This function is used only with a dialog sheet active.

Syntax

SET.DIALOG.DEFAULT(**object_id_text**)

Object_id_text   is the name of the button control to set as the default button, as in "Button 5".

Related Function

SET.DIALOG.FOCUS   Sets the focus of a dialog box

SET.DIALOG.FOCUS

Sets the focus of a dialog box. This function is used only with a dialog sheet active.

Syntax

SET.DIALOG.FOCUS(**object_id_text**)

Object_id_text   the name of the control or object as text to give the focus to, as in "Check box 4".

Related Function

SET.DIALOG.DEFAULT   Sets which button is automatically pressed (the default button) when the user presses ENTER

SET.EXTRACT

Equivalent to clicking the Set Extract command on the Data menu in Microsoft Excel version 4.0. Defines the name Extract for the selected range on the active sheet.

Syntax

SET.EXTRACT( )

Related Functions

SET.DATABASE   Equivalent to clicking the Set Database command on the Data menu in Microsoft Excel version 4.0

SET.CRITERIA   Equivalent to clicking the Set Criteria command on the Data menu in Microsoft Excel version 4.0

SET.LIST.ITEM

Sets the text of an item in a list box or drop-down box control.

Syntax

SET.LIST.ITEM(**text**, **index_num**)

Text   specifies the text of the item to be added. Instead of text, an empty string may be inserted.

Index_num   is the list index of the item to be changed, from 1 to the number of items in the list.

Remarks

If the list box or drop-down box was already filled using the LISTBOX.PROPERTIES function, then changing an item with SET.LIST.ITEM causes the fillrange contents to be discarded, leaving a list with one non-blank element and index_num entries.

Related Functions

REMOVE.LIST.ITEM   Removes an item in a list box or drop-down box

SELECT.LIST.ITEM   Selects an item in a list box or in a group box

SET.NAME

Defines a name on a macro sheet to refer to a value. The defined name exists only on the macro sheet's list of names and does not appear in the global list of names for the workbook. The SET.NAME function is useful for storing values while the macro is calculating.

Syntax

SET.NAME(**name_text**, value)

Name_text   is the name in the form of text that refers to value.

Value   is the value you want to store in name_text.

- If value is omitted, the name name_text is deleted.
- If value is a reference, name_text is defined to refer to that reference.

Remarks

- If you want to define a name as a constant value, you can use the following syntax instead of SET.NAME:

  name_text=value

  See the first two examples following.
- SET.NAME defines names as absolute references, even if a relative reference is specified. See the third and fourth examples following.
- If you want name_text to refer permanently to the value of a referenced cell rather than to the reference itself, you must use the DEREF function. Use of DEREF prevents name_text from referring to a new value every time the contents of the referenced cell changes. See the last example following.

Tips

- If you need to return an array to a macro sheet (for example, if the macro needs a list of all open windows), assign a name to the array instead of placing the array information in a range of cells. For example:

  `SET.NAME("OpenDocuments", WINDOWS())` or
  `SET.NAME("OpenDocuments", {"WORKSHEET1", "WORKSHEET2"})`
- You can then use the INDEX function with the name you have defined to access items in the array stored in the name.
- When you're debugging a macro and want to know the current value assigned to a name created by SET.NAME, you can halt the macro, click Define on the Name submenu of the Insert menu, and select the name from the Define Name dialog box.

Examples

Each of these formulas defines the name Counter to refer to the constant number 1 on the macro sheet:

```
SET.NAME("Counter", 1)
```

```
Counter=1
```

Each of these formulas redefines Counter to refer to the current value of Counter plus 1:

```
SET.NAME("Counter", Counter+1)
```

```
Counter=Counter+1
```

The following macro formula defines the name Reference to refer to cell $A$1:

```
SET.NAME("Reference", A1)
```

The following macro formula defines the name Results to refer to the cells $A$1:$C$3:

```
SET.NAME("Results", A1:C3)
```

The following macro formula defines the name Range as the current selection:

```
SET.NAME("Range", SELECTION())
```

If $A$1 contains the value 2, the following macro formula defines the name Index to refer to the constant value 2:

```
SET.NAME("Index", DEREF(A1))
```

Related Functions

DEFINE.NAME	Defines a name on the active worksheet or macro sheet
SET.VALUE	Sets the value of a cell on a macro sheet

SET.PAGE.BREAK

Equivalent to clicking the Page Break command on the Insert menu. Sets manual page breaks. Use SET.PAGE.BREAK to override the automatic page breaks. Setting a manual page break changes the automatic page breaks that follow it.

The page break occurs above and to the left of the active cell and appears as dotted lines if you have set up a printer. If the active cell is in column A, a manual page break is added only above the cell. If the active cell is in row 1, a manual page break is added only at the left edge of the cell. If the row or column next to the active cell already has a page break, SET.PAGE.BREAK takes no action.

Syntax

SET.PAGE.BREAK( )

Related Functions

PREVIEW Previews pages and page breaks before printing

REMOVE.PAGE.BREAK Removes manual page breaks

SET.PREFERRED

Changes the default format that Microsoft Excel uses when you create a new chart or when you format a chart PREFERRED macro function. When you use the SET.PREFERRED function, the format of the active chart becomes the preferred format.

Syntax

SET.PREFERRED(format)

Format is the name of the format that you want as the default format for charts. If omitted, the format of the currently active chart is used. If format is "Built_in", then Microsoft Excel will use the standard, built-in chart as the default. If the chart was created in Microsoft Excel version 4.0 and if format is "PREFERRED", then the preferred chart format used in Microsoft Excel version 4.0 will be used. Format is case sensitive.

Related Function

PREFERRED Changes the format of the active chart to the preferred format

SET.PRINT.AREA

Defines the print area for the workbook—the area that prints when you click the Print command on the File menu. Equivalent to entering a range in the Print Area edit box on the Sheet tab in the Page Setup dialog box, which appears when you click the Page Setup command on the File menu.

Syntax

SET.PRINT.AREA(range)

Range is the reference to the range that you want to be printed. If you specify no range by using a set of empty quotation marks (""), deletes the print area.

Remarks

- If you use SET.PRINT.AREA with a multiple selection and then use the PRINT function, the individual selections are printed one after the other in the order they were selected.
- To resume printing the entire worksheet, click the Page Setup command on the File menu and click the Sheet tab. Then delete the range in the Print Area edit box.

Related Functions

PRINT Prints the active sheet

SET.PRINT.TITLES Identifies text to print as titles

SET.PRINT.TITLES

Defines the print titles for the sheet. Use SET.PRINT.TITLES if you want Microsoft Excel to print the titles whenever it prints any cells in a row or column that intersect the print titles area; a cell need only share the row or column with a print title for the title to be printed above or to the left of that cell.

Syntax

SET.PRINT.TITLES(titles_for_cols_ref, titles_for_rows_ref)

SET.PRINT.TITLES?(titles_for_cols_ref, titles_for_rows_ref)

Titles_for_cols_ref   is a reference to the row to be used as a title for columns.

- If you specify part of a row, Microsoft Excel expands the title to a full row.
- If you omit titles_for_cols_ref, Microsoft Excel uses the existing row of column titles, if any.
- If you specify empty text (""), Microsoft Excel removes the row from the print titles definition.

Titles_for_rows_ref   is a reference to the column to be used as a title for rows.

- If you specify part of a column, Microsoft Excel expands the title to a full column.
- If you omit titles_for_rows_ref, Microsoft Excel uses the existing column of row titles, if any.
- If you specify empty text (""), Microsoft Excel removes the column from the print titles definition.

Remarks

- SET.PRINT.TITLES operates on the current sheet. If you specify a range that is invalid for the current sheet, Microsoft Excel returns the #VALUE error value.
- The print titles selection can be a multiple selection. Microsoft Excel names this selection Print_Titles when SET.PRINT.TITLES is run.

Related Functions

DEFINE.NAME   Defines a name on the active worksheet or macro sheet

PRINT   Prints the active sheet

SET.PRINT.AREA   Defines the print area

SET.UPDATE.STATUS

Sets the update status of a link to automatic or manual. Use SET.UPDATE.STATUS to change the way a link is updated.

Syntax

SET.UPDATE.STATUS(**link_text, status**, type_of_link)

Link_text is the path of the linked file for which you want to change the update status.

Status is the number 1 or 2 and describes how you want the link to be updated.

Status	Update method
1	Automatic
2	Manual

Type_of_link is a number from 1 to 4 that specifies what type of link you want to get information about.

Type_of_link	Link document type
1	Not available
2	DDE/OLE link
3	Not available
4	Not available

Example

In Microsoft Excel for Windows, the following macro formula sets the update status of the DDE link to Microsoft Word for Windows to manual:

```
SET.UPDATE.STATUS("WordDocument|'C:\MEMO.DOC'!DDE.LINK1", 2, 2)
```

Related Functions

GET.LINK.INFO Returns information about a link

UPDATE.LINK Updates a link to another document

SET.VALUE

Changes the value of a cell or cells on the macro sheet (not the worksheet) without changing any formulas entered in those cells. Use SET.VALUE to assign initial values and to store values during the calculation of a macro. SET.VALUE is especially useful for initializing a dialog box and the conditional test in a WHILE loop. SET.VALUE assigns values to a specific reference or to the name of a reference that has already been defined. For information about creating a new name or entering data on a worksheet, see "Remarks" later in this topic.

Syntax

SET.VALUE(**reference**, values)

Reference specifies the cell or cells on the macro sheet to which you want to assign a new value or values. If the cell is empty, enters the value in the cell.

- If a cell in reference previously contained a formula, the formula is not changed, but the value of the cell might change. See the second example following.
- If reference is a reference to a range of cells, rather than to a single cell, then values should be an array of the same size. If not, Microsoft Excel expands it into multiple values using the normal rules for expanding arrays. See the third example following.

Values is the value or set of values to which you want to assign the cell or cells in reference.

Remarks

Consider the following guidelines as you choose a function to set values on a worksheet or macro sheet:

- Use SET.VALUE to assign initial values to a reference (including names that have already been defined) on a macro sheet, and to store values during the calculation of a macro.
- Use FORMULA to enter values in a worksheet cell.
- Use SET.NAME to change the value of a name on a macro sheet (the name is created if it does not already exist). For more information, see SET.NAME.
- Use DEFINE.NAME to create or change the value of a name on a worksheet.

Examples

The following macro formula changes the value of cell A1 on the macro sheet to 1:

```
SET.VALUE($A$1, 1)
```

Suppose the name TempAverage refers to a cell containing the formula AVERAGE(Temp1, Temp2, Temp3). The following formula assigns the value 99 to this cell, even if the average of the arguments is not 99, without changing the formula in TempAverage:

```
SET.VALUE(TempAverage, 99)
```

The preceding formula is useful if a WHILE loop or some other conditional test depends on TempAverage and you want to force the conditional test to have a particular result. Of course, TempAverage is restored to its correct value as soon as it is recalculated. (Recall that unlike formulas in a worksheet, formulas in a macro sheet are not recalculated until the macro actually uses them.)

The following macro formula stores the values 1, 2, 3, and 4 in cells A1:B2:

```
SET.VALUE($A$1:$B$2, {1, 2;3, 4})
```

Related Functions

DEFINE.NAME   Defines a name on the active worksheet or macro sheet

FORMULA   Enters values into a cell or range or onto a chart

SET.NAME   Defines a name as a value

SHORT.MENUS

Equivalent to clicking the Short Menus command on the Options menu or the Chart menu in Microsoft Excel version 3.0 or earlier.

Syntax

SHORT.MENUS(logical)

SHOW.ACTIVE.CELL

Scrolls the active window so the active cell becomes visible. If an object is selected, SHOW.ACTIVE.CELL returns the #VALUE! error value and halts the macro.

Syntax

SHOW.ACTIVE.CELL( )

Related Functions

ACTIVE.CELL   Returns the reference of the active cell

FORMULA.GOTO   Selects a named area or reference on any open workbook

SHOW.BAR

Displays the specified menu bar. Use SHOW.BAR to display a menu bar you have created with the ADD.BAR function or to display a built-in Microsoft Excel 95 or earlier version menu bar.

Syntax

SHOW.BAR(bar_num)

Bar_num   is the number of the menu bar you want to display. It can be the number of one of the Microsoft Excel built-in menu bars, the number returned by a previously executed ADD.BAR function, or a reference to a cell containing a previously executed ADD.BAR function.

If bar_num is omitted, Microsoft Excel displays the appropriate menu bar for the active workbook as shown in the following table.

Bar_num	Bar displayed
1	A sheet or macro sheet (Microsoft Excel version 4.0)
2	A chart (Microsoft Excel version 4.0)
3	No active window
4	The Info window (Microsoft Excel 95 or earlier versions)
5	A sheet or macro sheet (short menus)
6	A chart (short menus)
7	Shortcut menus 1 (for Cells, Workbook tabs, Toolbars, VB Windows)
8	Shortcut menus 2 (for objects)
9	Shortcut menus 3 (for chart elements)
10	A sheet or macro sheet
11	A chart
12	A Visual Basic module
13-35	Reserved for use by shortcut menus. These numbers will return an error if a macro tries to do anything with them.
37-51	Custom menu bar for macro use

Remarks

- When displaying a built-in menu bar, you can display only bars 1 or 5 if a sheet or macro sheet is active, bars 2 or 6 if a chart is active, and so on. If you try to display a chart menu bar while a sheet or macro sheet is active, SHOW.BAR returns an error and interrupts the current macro.
- Displaying a custom menu bar disables automatic menu-bar switching when different types of sheets are selected. For example, if a custom menu bar is displayed and you switch to a chart, neither of the two chart menus is automatically displayed as it would be when you are using the built-in menu bars. Automatic menu-bar switching is reenabled when a built-in bar is displayed using SHOW.BAR.

Example

The following macro formula displays short menus on a worksheet or macro sheet:

```
SHOW.BAR(5)
```

Related Functions

ADD.BAR   Adds a menu bar

DELETE.BAR   Deletes a menu bar

SHOW.TOOLBAR   Hides or displays a toolbar

SHOW.CLIPBOARD

Displays the contents of the Clipboard in a new window.

Syntax

SHOW.CLIPBOARD( )

Remarks

- In Microsoft Excel for Windows, the Clipboard must already be running if you want to display its contents in a new window. If it is not already running, you must run the SHOW.CLIPBOARD function twice, once to start the Clipboard application and again to display it in a new window.
- If the Clipboard contains cells, the window shows the size of the Clipboard contents in rows and columns. If the Clipboard contains text cut from the formula bar, the window displays the text.

SHOW.DETAIL

Expands or collapses the detail under the specified expand or collapse button.

Syntax

SHOW.DETAIL(**rowcol, rowcol_num**, expand, show_field)

Rowcol   is a number that specifies whether to operate on rows or columns of data.

Rowcol	Operates on
1	Rows
2	Columns
3	The current cell's row or column. The second argument, rowcol_num, is then ignored.

Rowcol_num is a number that specifies the row or column to expand or collapse. If you are in A1 mode, you must still give the column as a number. If rowcol_num is not a summary row or column, SHOW.DETAIL returns the #VALUE! error value and interrupts the macro.

Expand is a logical value that specifies whether to expand or collapse the detail under the row or column. If expand is TRUE, Microsoft Excel expands the detail under the row or column; if FALSE, it collapses the detail under the row or column. If expand is omitted, the detail is expanded if it is currently collapsed and collapsed if it is currently expanded.

Show_Field is a string specifying the name of the field to add to a PivotTable report, if the selection is inside a PivotTable report. The new field is added as the new innermost field. Available for only innermost row or column fields.

Related Function

SHOW.LEVELS Displays a specific number of levels of an outline

SHOW.DIALOG

Runs a dialog on a dialog sheet.

Syntax

SHOW.DIALOG(dialog_sheet)

Dialog_sheet is the name of the dialog sheet to run. If omitted, the active sheet will be the sheet that is run. If this function is run on a sheet other than a dialog sheet, this function returns the #VALUE error value.

Remarks

Returns TRUE if the dialog box is closed by the user choosing an OK button. Returns FALSE if the dialog box is cancelled by choosing the Cancel button or the ESC key, or in Microsoft Excel for the Macintosh by pressing COMMAND+. (period).

Related Function

HIDE.DIALOG Closes the dialog box that has the current focus

SHOW.INFO

This function should not be used. The Info Window has been removed from Microsoft Excel 97 or later.

Related Functions

FORMULA.GOTO Selects a named area or reference on any open workbook

GET.CELL Returns information about the specified cell

SELECT Selects a cell, worksheet object, or chart item

SHOW.LEVELS

Displays the specified number of row and column levels of an outline.

Syntax

SHOW.LEVELS(row_level, col_level)

Row_level specifies the number of row levels of an outline to display. If the outline has fewer levels than specified by row_level, Microsoft Excel shows all levels. If row_level is zero or omitted, no action is taken on rows.

Col_level specifies the number of column levels of an outline to display. If the outline has fewer levels than specified by col_level, Microsoft Excel shows all levels. If col_level is zero or omitted, no action is taken on columns.

Remarks

If you omit both arguments, SHOW.LEVELS returns the #VALUE! error value.

Related Function

SHOW.DETAIL Expands or collapses a portion of an outline

SHOW.TOOLBAR

Equivalent to selecting the check box corresponding to a toolbar on the Toolbars tab in the Customize dialog box, which appears when you select the Customize command (View menu, Toolbars submenu). Hides or displays a toolbar. Use SHOW.TOOLBAR to display or hide a menu bar you have created with the ADD.BAR function or to display a built-in Microsoft Excel 95 or earlier version toolbar.

Syntax

SHOW.TOOLBAR(**bar_id, visible**, dock, x_pos, y_pos, width, protect, tool_tips, large_buttons, color_buttons)

Bar_id is a number or name of a toolbar corresponding to the toolbars you want to display. For detailed information about bar_id, see ADD.TOOL.

Visible is a logical value that, if TRUE, specifies that the toolbar is visible or, if FALSE, specifies that the toolbar is hidden.

Dock specifies the docking location of the toolbar.

Dock	Position of toolbar
1	Top of workspace
2	Left edge of workspace
3	Right edge of workspace
4	Bottom of workspace
5	Floating (not docked)

X_pos specifies the horizontal position of the toolbar.

- If the toolbar is docked (not floating), x_pos is measured horizontally from the left edge of the toolbar to the left edge of the toolbar's docking area.

- If the toolbar is floating, x_pos is measured horizontally from the left edge of the toolbar to the right edge of the rightmost toolbar in the left docking area.
- X_pos is measured in points. A point is 1/72nd of an inch.

Y_pos specifies the vertical position of the toolbar.

- If the toolbar is docked, y_pos is measured vertically from the top edge of the toolbar to the top edge of the toolbar's docking area.
- If the toolbar is floating, y_pos is measured vertically from the top edge of the toolbar to the top edge of the Microsoft Excel workspace.
- Y_pos is measured in points.

Width specifies the width of the toolbar and is measured in points. If you omit width, Microsoft Excel uses the existing width setting.

Protect is a number specifying the degree to which you can modify a toolbar and its buttons. Each succeeding protect number retains the protection status of its previous numbers. For example, a protect status of 3 (a toolbar cannot become docked if it is floating) assumes the protection status of 0, 1, and 2 as well.

Protect	Description
0	Default. Toolbars can be re-shaped, docked, and floating. Toolbar buttons can be removed from and moved to the toolbar.
1	Toolbars can be re-shaped, docked, and floating. Toolbar buttons can not be removed from nor moved to the toolbar.
2	A floating toolbar cannot be re-shaped. It can be docked.
3	A floating toolbar cannot be docked. If it is already docked, it cannot become floating.
4	The toolbar cannot be moved at all. If it is already floating, it cannot be re-shaped or moved. If it is docked, it cannot become un-docked.

Tool_tips is a logical value that corresponds to the Show Screentips On Toolbars check box on the Options tab. If TRUE, ScreenTips will be displayed. If FALSE, ScreenTips will not be displayed.

Large Buttons is a logical value that corresponds to the Large Icons check box on the Options tab. If TRUE, large icons will be displayed. If FALSE, large icons will not be displayed.

Color_buttons is a logical value that corresponds to the Color Toolbars check box. If TRUE, the toolbar buttons will be displayed in color. If FALSE, the toolbar buttons will not be displayed in color. This argument is for compatibility with Microsoft Excel version 5.0.

Related Functions

ADD.BAR   Adds a menu bar

ADD.TOOLBAR   Creates a new toolbar with the specified tools

SIZE

Equivalent to clicking the Size command on the Control menu in Microsoft Excel for Windows version 3.0 or earlier or to changing the size of a window by dragging its border. In Microsoft Excel for the Macintosh version 3.0 or earlier, equivalent to changing the size of a window by dragging its size box. This function is included only for macro compatibility and will be converted to WINDOW.SIZE when you open older macro sheets. For more information, see WINDOW.SIZE.

Syntax

SIZE(**width,height**,window_text)

SIZE?(width,height,window_text)

Related Function

WINDOW.SIZE   Changes the size of the active window

SLIDE.COPY.ROW

This function should not be used in Microsoft Excel 95 or later because the Slide Show add-in is available only in Microsoft Excel version 5.0 or earlier versions.

Equivalent to clicking the Copy Row button on a slide show sheet. Copies the selected slides, each of which is defined on a single row, to the Clipboard.

If this function is not available, you must install the Slide Show add-in.

Syntax

SLIDE.COPY.ROW( )

Remarks

- SLIDE.COPY.ROW, SLIDE.CUT.ROW, SLIDE.DELETE.ROW, and SLIDE.PASTE.ROW return TRUE if successful, or FALSE if not successful. If the active sheet is not a slide show or is protected, these functions return the #N/A error value. If the current selection is not valid, these functions return the #VALUE! error value.

Related Functions

SLIDE.CUT.ROW   Cuts the selected slides and pastes them onto the Clipboard

SLIDE.DEFAULTS   Specifies default values for the active slide show sheet

SLIDE.DELETE.ROW   Deletes the selected slides

SLIDE.EDIT   Changes the attributes of the selected slide

SLIDE.GET   Returns information about a slide or slide show

SLIDE.PASTE   Pastes the contents of the Clipboard onto a slide

SLIDE.PASTE.ROW   Pastes previously cut or copied slides onto the current selection

SLIDE.SHOW   Starts a slide show in the active sheet

SLIDE.CUT.ROW

This function should not be used in Microsoft Excel 95 or later because the Slide Show add-in is available only in Microsoft Excel version 5.0 or earlier versions.

Equivalent to clicking the Cut Row button on a slide show sheet. Cuts the selected slides, each of which is defined on a single row, and pastes them onto the Clipboard. For more information, see SLIDE.COPY.ROW.

If this function is not available, you must install the Slide Show add-in.

Syntax

SLIDE.CUT.ROW( )

Related Function

SLIDE.COPY.ROW   Copies the selected slides and pastes them onto the Clipboard

SLIDE.DEFAULTS

This function should not be used in Microsoft Excel 95 or later because the Slide Show add-in is available only in Microsoft Excel version 5.0 or earlier versions.

Equivalent to clicking the Set Defaults button on a slide show sheet. Specifies the default values for the transition effect, speed, advance rate, and sound on the active slide show sheet.

If this function is not available, you must install the Slide Show add-in.

Syntax

SLIDE.DEFAULTS(effect_num, speed_num, advance_rate_num, soundfile_text)

SLIDE.DEFAULTS?(effect_num, speed_num, advance_rate_num, soundfile_text)

For a description of the arguments, see SLIDE.PASTE. If an argument is omitted, its default value is not changed.

Remarks

- SLIDE.DEFAULTS returns TRUE if it successfully changes the default values, or FALSE if you click the Cancel button when using the dialog-box form. If the active sheet is not a slide show or is protected, SLIDE.DEFAULTS returns the #N/A error value.

SLIDE.DELETE.ROW

This function should not be used in Microsoft Excel 95 or later because the Slide Show add-in is available only in Microsoft Excel version 5.0 or earlier versions.

Equivalent to clicking the Delete Row button on a slide show sheet. Deletes the selected slides, each of which is defined on a single row. For more information, see SLIDE.COPY.ROW.

If this function is not available, you must install the Slide Show add-in.

Syntax

SLIDE.DELETE.ROW( )

Related Function

SLIDE.COPY.ROW Copies the selected slides and pastes them onto the Clipboard

SLIDE.EDIT

This function should not be used in Microsoft Excel 95 or later because the Slide Show add-in is available only in Microsoft Excel version 5.0 or earlier versions.

Equivalent to clicking the Edit button in a slide show sheet. Gives the currently selected slide the attributes you specify.

If this function is not available, you must install the Slide Show add-in.

Syntax

SLIDE.EDIT(effect_num, speed_num, advance_rate_num, soundfile_text)

SLIDE.EDIT?(effect_num, speed_num, advance_rate_num, soundfile_text)

For a description of the arguments, see SLIDE.PASTE.

Remarks

- SLIDE.EDIT returns TRUE if it successfully edits the slide, or FALSE if you click the Cancel button when using the dialog-box form. If the active sheet is not a slide show or is protected, SLIDE.EDIT returns the #N/A error value. If the current selection is not a valid slide, SLIDE.EDIT returns the #VALUE error value.

Related Function

SLIDE.PASTE Pastes the contents of the Clipboard onto a slide

SLIDE.GET

This function should not be used in Microsoft Excel 95 or later because the Slide Show add-in is available only in Microsoft Excel version 5.0 or earlier versions.

Returns the specified information about a slide show or a specific slide in the slide show.

If this function is not available, you must install the Slide Show add-in.

Syntax

SLIDE.GET(**type_num**, name_text, slide_num)

Type_num is a number specifying the type of information you want.

These values of type_num return information about a slide show.

Type_num	**Type of information**
1	Number of slides in the slide show

2	A two-item horizontal array containing the numbers of the first and last slides in the current selection, or the #VALUE error value if the selection is nonadjacent
3	Version number of the Slide Show add-in that created the slide show sheet

These values of type_num return information about a specific slide in the slide show.

Type_num	**Type of information**
4	Transition effect number
5	Transition effect name
6	Transition effect speed
7	Number of seconds the slide is displayed before advancing
8	Name of the sound file associated with the slide, or empty text ("") if none is specified (in Microsoft Excel for the Macintosh, this includes the number or name of the sound resource within the sound file)

Name_text is the name of an open slide show sheet for which you want information. If name_text is omitted, it is assumed to be the active sheet.

Slide_num is the number of the slide about which you want information.

- If slide_num is omitted, it is assumed to be the slide associated with the active cell on the sheet specified by name_text.
- If type_num is 1 through 3, slide_num is ignored.

SLIDE.PASTE

This function should not be used in Microsoft Excel 95 or later because the Slide Show add-in is available only in Microsoft Excel version 5.0 or earlier versions.

Equivalent to clicking the Paste button on a slide show sheet. Pastes the contents of the Clipboard as the next available slide of the active slide show sheet, and gives the slide the attributes you specify.

If this function is not available, you must install the Slide Show add-in.

Syntax

SLIDE.PASTE(effect_num, speed_num, advance_rate_num, soundfile_text)

SLIDE.PASTE?(effect_num, speed_num, advance_rate_num, soundfile_text)

Effect_num is a number specifying the transition effect you want to use when displaying the slide.

- The numbers correspond to the effects in the Effect list in the Edit Slide dialog box. The first effect in the list is 1 (None).
- If effect_num is omitted, the default setting is used.

Speed_num is a number from 1 to 10 specifying the speed of the transition effect.

- If speed_num is omitted, the default setting is used.
- If speed_num is greater than 10, Microsoft Excel uses the value 10 anyway.
- If effect_num is 1 (none), speed_num is ignored.

Advance_rate_num is a number specifying how long (in seconds) the slide is displayed before advancing to the next one.

- If advance_rate_num is omitted, the default setting is used.
- If advance_rate_num is 0, you must press a key or click with the mouse to advance to the next slide.

Soundfile_text is the name of a file enclosed in quotation marks and specifies sound that will be played when the slide is displayed.

- If soundfile_text is omitted, Microsoft Excel plays the default sound defined for the slide show sheet, if any.
- If soundfile_text is empty text (""), no sound is played.
- In Microsoft Excel for the Macintosh, soundfile_text also includes the number or name of the sound resource to play in the file.

Resource is the number or name of a sound resource in soundfile_text.

- This argument applies only to Microsoft Excel for the Macintosh.
- If resource is omitted, Microsoft Excel uses the first resource in the file.
- If the file does not contain a sound resource with the specified name or number, Microsoft Excel halts the macro and displays an error message.

Remarks

- SLIDE.PASTE returns TRUE if it successfully pastes the slide, or FALSE if you click the Cancel button when using the dialog-box form. If the active sheet is not a slide show or is protected, SLIDE.PASTE returns the #N/A error value. If the Clipboard format is not compatible with the slide show sheet's format, SLIDE.PASTE returns the #VALUE error value.

Examples

In Microsoft Excel for Windows, the following macro formula pastes the contents of the Clipboard into the active slide show sheet. The slide's transition effect is fade, at a speed of 8; it is displayed for five seconds; and Microsoft Excel plays the specified sound file:

```
SLIDE.PASTE(3, 8, 5, "C:\SLIDES\SOUND\MACHINES.WAV")
```

In Microsoft Excel for the Macintosh, the formula is:

```
SLIDE.PASTE(3, 8, 5, "HARD DISK:SLIDES:SOUND:MACHINE SOUNDS")
```

SLIDE.PASTE.ROW

This function should not be used in Microsoft Excel 95 or later because the Slide Show add-in is available only in Microsoft Excel version 5.0 or earlier versions.

Equivalent to clicking the Paste Row button on a slide show sheet. Pastes previously cut or copied slides onto the current selection. For more information, see SLIDE.COPY.ROW.

If this function is not available, you must install the Slide Show add-in.

Syntax

SLIDE.PASTE.ROW( )

Related Function

SLIDE.COPY.ROW   Copies the selected slides and pastes them onto the Clipboard

SLIDE.SHOW

This function should not be used in Microsoft Excel 95 or later because the Slide Show add-in is available only in Microsoft Excel version 5.0 or earlier versions.

Equivalent to clicking the Start Show button on a slide show sheet. Starts the slide show in the active sheet.

If this function is not available, you must install the Slide Show add-in.

Syntax

SLIDE.SHOW(initialslide_num, repeat_logical, dialogtitle_text, allownav_logical, allowcontrol_logical)

SLIDE.SHOW?(initialslide_num, repeat_logical, dialogtitle_text, allownav_logical, allowcontrol_logical)

All arguments except dialogtitle_text correspond to options and settings in the Start Show dialog box.

Initialslide_num   is a number from 1 to the number of slides in the slide show and specifies which slide to display first. If omitted, it is assumed to be 1.

Repeat_logical   is a logical value specifying whether to repeat or end the slide show after displaying the last slide. If repeat_logical is TRUE, the slide show repeats; if FALSE or omitted, the slide show ends.

Dialogtitle_text   is text enclosed in quotation marks that specifies the title of the dialog boxes displayed during the slide show. If dialogtitle_text is omitted, it is assumed to be "Slide Show".

Allownav_logical   is a logical value specifying whether to enable or disable navigational keys (arrow keys, PAGE UP, PAGE DOWN, and so on) or the mouse during the slide show. If allownav_logical is TRUE or omitted, you can press navigational keys or use the mouse to move between slides; if FALSE, all movement is controlled by the slide show sheet settings.

Allowcontrol_logical   is a logical value specifying whether to enable or disable the Slide Show Options dialog box during the slide show. If allowcontrol_logical is TRUE or omitted, you can press ESC to interrupt the slide show and display the dialog box; if FALSE, pressing ESC stops the slide show but does not display the dialog box.

Tip   If you want to display the last slide in a show but don't know its number, use SLIDE.GET(1) as the initialslide_num argument.

Remarks

SLIDE.SHOW returns the values shown in the following table:

Situation	**Returned value**
The slide show ends normally.	TRUE
You press the Cancel button when using the dialog-box form.	FALSE
The active sheet is not a slide show or is protected.	#N/A
You interrupt the slide show, and then stop it.	1

SOLVER.ADD

Equivalent to clicking the Solver command on the Tools menu and clicking the Add button in the Solver Parameters dialog box. Adds a constraint to the current problem. For an explanation of constraints, see "Remarks" later in this topic.

If this function is not available, you must install the Solver add-in.

Syntax

SOLVER.ADD(**cell_ref, relation**, formula)

Cell_ref   is a reference to a cell or range of cells on the active sheet and forms the left side of the constraint.

Relation   specifies the arithmetic relationship between the left and right sides, or whether cell_ref must be an integer.

Relation	Arithmetic relationship
1	<=
2	=
3	>=
4	Int (cell_ref is an integer)

Formula is the right side of the constraint and will often be a single number, but it may be a formula (as text) or a reference to a range of cells.

- If relation is 4, cell_ref must be a subset of the references in the By Changing cells text box.
- if relation is 4, formula must be either "=integer" or "integer".
- Any cell reference in a formula must use the R1C1 reference style.
- If formula is a reference to a range of cells, the number of cells in the range usually matches the number of cells in cell_ref, although the shape of the areas need not be the same. For example, cell_ref could be a row and formula could refer to a column, as long as the number of cells is the same. Formula can also be a single reference, as in the following relationship: A1:A4 <= B1.

Remarks

- The SOLVER.ADD, SOLVER.CHANGE, and SOLVER.DELETE functions correspond to the Add, Change, and Delete buttons in the Solver Parameters dialog box. You use these functions to define constraints. For many macro applications, however, you may find it more convenient to load the problem specifications from the sheet in a single step using the SOLVER.LOAD function.
- Each constraint is uniquely identified by the combination of the cell reference on the left and the relationship (<=, =, or >=) between its left and right sides, or the cell reference may be defined as an integer only. This takes the place of selecting the appropriate constraint in the Tools Solver Parameters dialog box. You can manipulate the constraints with SOLVER.CHANGE or SOLVER.DELETE. The constraints in a Solver problem can refer to a maximum of 400 cells.

Related Function

SOLVER.DELETE Deletes an existing constraint

SOLVER.CHANGE

Equivalent to clicking the Solver command on the Tools menu and clicking the Change button in the Solver Parameters dialog box. Changes the right side of an existing constraint.

If this function is not available, you must install the Solver add-in.

Syntax

SOLVER.CHANGE(**cell_ref, relation**, formula)

For an explanation of the arguments and constraints, see SOLVER.ADD.

Remarks

- If the combination of cell_ref and relation does not match any existing constraint, the function returns the value 4 and no action is taken.
- To change the cell_ref or relation of an existing constraint, use SOLVER.DELETE to delete the old constraint and then use SOLVER.ADD to add the constraint in the form you want.

Related Functions

SOLVER.DELETE	Deletes an existing constraint
SOLVER.ADD	Adds a constraint to the current problem

SOLVER.DELETE

Equivalent to clicking the Solver command on the Tools menu and clicking the Delete button in the Solver Parameters dialog box. Deletes an existing constraint.

If this function is not available, you must install the Solver add-in.

Syntax

SOLVER.DELETE(**cell_ref, relation**, formula)

For an explanation of the arguments and constraints, see SOLVER.ADD.

Remarks

If the combination of cell_ref and relation does not match any existing constraint, the function returns the value 4 and no action is taken. If the constraint is found, it is deleted, and the function returns the value 0.

Related Function

SOLVER.ADD	Adds a constraint to the current problem

SOLVER.FINISH

Equivalent to clicking OK in the Solver Results dialog box that appears when the solution process is complete. The dialog-box form displays the dialog box with the arguments that you supply as defaults. This function must be used if you supplied the value TRUE for the userfinish argument to SOLVER.SOLVE.

If this function is not available, you must install the Solver add-in.

Syntax

SOLVER.FINISH(keep_final, report_array)

SOLVER.FINISH?(keep_final, report_array)

Keep_final is the number 1 or 2 and specifies whether to keep the final solution. If keep_final is 1 or omitted, the final solution values are kept in the changing cells. If keep_final is 2, the final solution values are discarded and the former values of the changing cells are restored.

Report_array is an array argument specifying what reports to create when Solver is finished.

If report_array is	Microsoft Excel creates
{1}	An answer report
{2}	A sensitivity report
{3}	A limit report

Any combination of these produces multiple reports. For example, if report_array is {1, 2}, Microsoft Excel creates an answer report and a sensitivity report.

Related Function

SOLVER.SOLVE Equivalent to clicking the Solver command on the Tools menu and clicking the Solve button in the Solver Parameters dialog box

SOLVER.GET

Returns information about current settings for Solver. The settings are specified in the Solver Parameters and Solver Options dialog boxes.

If this function is not available, you must install the Solver add-in.

Syntax

SOLVER.GET(**type_num**, sheet_name)

Type_num is a number specifying the type of information you want.

The following settings are specified in the Solver Parameters dialog box.

Type_Num	Returns
2	A number corresponding to the Equal To option 1 = Max 2 = Min 3 = Value of
3	The value in the Value Of box

4	The reference (as a multiple reference if necessary) in the By Changing Cells box
5	The number of constraints
6	An array of the left sides of the constraints in the form of text
7	An array of numbers corresponding to the relationships between the left and right sides of the constraints: 1 = <= 2 = = 3 = >= 4 = int
8	An array of the right sides of the constraints in the form of text

The following settings are specified in the Solver Options dialog box:

Type_Num	**Returns**
10	The maximum number of iterations
11	The precision
12	The integer tolerance value
13	TRUE if the Assume Linear Model check box is selected; FALSE otherwise
14	TRUE if the Show Iteration Results check box is selected; FALSE otherwise
15	TRUE if the Use Automatic Scaling check box is selected; FALSE otherwise
16	A number corresponding to the type of estimates: 1 = Tangent 2 = Quadratic
17	A number corresponding to the type of derivatives: 1 = Forward 2 = Central

18	A number corresponding to the type of search: 1 = Quasi-Newton 2 = Conjugate Gradient

Sheet_name is the name of a sheet that contains the scenario for which you want information. If sheet_name is omitted, it is assumed to be the active sheet.

SOLVER.LOAD

Equivalent to clicking the Solver command on the Tools menu, clicking the Options button in the Solver Parameters dialog box, and clicking the Load Model button in the Solver Options dialog box. Loads Solver problem specifications that you have previously saved on the worksheet.

If this function is not available, you must install the Solver add-in.

Syntax

SOLVER.LOAD(**load_area**)

Load_area is a reference on the active sheet to a range of cells from which you want to load a complete problem specification.

- The first cell in load_area contains a formula for the Set Cell box; the second cell contains a formula for the changing cells; subsequent cells contain constraints in the form of logical formulas. The last cell optionally contains an array of Solver option values. The order of the Solver option values is the same as the top-to-bottom order in the Solver Options dialog box.
- Although load_area must be on the active sheet, it need not be the current selection.

SOLVER.OK

Equivalent to clicking the Solver command on the Tools menu and specifying options in the Solver Parameters dialog box. Specifies basic Solver options, except that constraints are added via SOLVER.ADD.

If this function is not available, you must install the Solver add-in.

Syntax

SOLVER.OK(set_cell, max_min_val, value_of, **by_changing**)

SOLVER.OK?(set_cell, max_min_val, value_of, by_changing)

Set_cell corresponds to the Set Target Cell box in the Solver Parameters dialog box.

- Set_cell must be a reference to a cell on the active worksheet.
- If you enter a cell reference, you must also enter a value for max_min_val. If you do not enter a cell, you must include three commas before the by_changing value.

Max_min_val corresponds to the options Max, Min, and Value Of in the Solver Parameters dialog box. Use this option only if you entered a reference for set_cell.

Max_min_val	Option specified
1	Maximize
2	Minimize
3	Match specific value

Value_of is a number that becomes the target for the cell in the Set Target Cell box if max_min_val is 3. Value_of is ignored if the cell is being maximized or minimized.

By_changing indicates the changing cells, as entered in the By Changing Cells box. By_changing must refer to a cell or range of cells on the active worksheet, and can be a multiple selection.

Remarks

The constraints in a Solver problem can refer to a maximum of 400 cells.

Related Function

SOLVER.SOLVE Returns an integer value indicating the condition that caused Solver to stop

SOLVER.OPTIONS

Equivalent to clicking the Solver command on the Tools menu and then clicking the Options button in the Solver Parameters dialog box. Specifies the available options.

If this function is not available, you must install the Solver add-in.

Syntax

SOLVER.OPTIONS(max_time, iterations, precision, assume_linear, step_thru, estimates, derivatives, search, int_tolerance, scaling)

The arguments correspond to the options in the dialog box. If an argument is omitted, Microsoft Excel uses an appropriate value based on the current situation. If any of the arguments are the wrong type, the function returns the #N/A error value. If an argument has the correct type but an invalid value, the function returns a positive integer corresponding to its position. A zero indicates all options were accepted.

Max_time must be an integer greater than zero and less than 32768. It corresponds to the Max Time box.

Iterations must be an integer greater than zero and less than 32768. It corresponds to the Iterations box.

Precision must be a number between zero and one, but not equal to zero or one. It corresponds to the Precision box.

Assume_linear is a logical value corresponding to the Assume Linear Model check box and allows Solver to arrive at a solution more quickly. If TRUE, Solver assumes that the underlying model is linear; if FALSE, it does not.

Step_thru is a logical value corresponding to the Show Iteration Results check box. If you have supplied SOLVER.SOLVE with a valid command macro reference, your macro will be called each time Solver pauses. If TRUE, Solver pauses at each trial solution; if FALSE, it does not.

Estimates is the number 1 or 2 and corresponds to the Estimates options: 1 for the Tangent option and 2 for the Quadratic option.

Derivatives is the number 1 or 2 and corresponds to the Derivatives options: 1 for the Forward option and 2 for the Central option.

Search is the number 1 or 2 and corresponds to the Search options: 1 for the Quasi-Newton option and 2 for the Conjugate Gradient option.

Int_tolerance is a decimal number corresponding to the Tolerance box in the Solver Options dialog box, and must be between zero and 1, inclusively. This argument applies only if integer constraints have been defined.

Scaling is a logical value corresponding to the Use Automatic Scaling check box. If scaling is TRUE, then if two or more constraints differ by several orders of magnitude, Solver scales the constraints to similar orders of magnitude during computation. If scaling is FALSE, Solver calculates normally.

SOLVER.RESET

Equivalent to clicking the Solver command on the Tools menu and clicking the Reset All button in the Solver Parameters dialog box. Erases all cell selections and constraints from the Solver Parameters dialog box and restores all the settings in the Solver Options dialog box to their defaults.

If this function is not available, you must install the Solver add-in.

Syntax

SOLVER.RESET( )

SOLVER.SAVE

Equivalent to clicking the Solver command on the Tools menu, clicking the Options button in the Solver Parameters dialog box, and clicking the Save Model button in the Solver Options dialog box. Saves the Solver problem specifications on the worksheet.

If this function is not available, you must install the Solver add-in.

Syntax

SOLVER.SAVE (**save_area**)

Save_area is a reference on the active sheet to a range of cells or to the upper-left corner of a range of cells into which you want to paste the current problem specification.

- If you specify only one cell for save_area, the area is extended downwards for as many cells as are required to hold the problem specifications (3 plus the number of constraints).

- If you specify more than one cell and if the area is too small, the last constraints (in alphabetic order by cell reference) or options will be omitted and the function will return a nonzero value.

- Save_area must be on the active worksheet, but it need not be the current selection.

SOLVER.SOLVE

Equivalent to clicking the Solver command on the Tools menu and clicking the Solve button in the Solver Parameters dialog box. If successful, returns an integer value indicating the condition that caused Solver to stop as described in "Remarks" later in this topic.

If this function is not available, you must install the Solver add-in.

Syntax

SOLVER.SOLVE(user_finish, show_ref)

User_finish   is a logical value specifying whether to display the Solver Results dialog box.

- If user_finish is TRUE, SOLVER.SOLVE returns its integer value without displaying anything. Your macro should decide what action to take (for example, by examining the return value or presenting its own dialog box); it must call SOLVER.FINISH in any case to restore the sheet to its proper state.
- If user_finish is FALSE or omitted, Solver displays the Solver Results dialog box, which allows you to keep or discard the final solution and run reports.

Show_ref   is a macro to be called in place of the Show Trial Solution dialog box. It is used when you want to regain control whenever Solver finds a new intermediate solution value.

- For this argument to have an effect, the Show Iteration Results check box must be selected in the Solver Options dialog box. This can be done manually by selecting the check box, or automatically by calling SOLVER.OPTIONS in your macro.
- The macro you call can inspect the current solution values on the sheet or take other actions such as saving or charting the intermediate values. It must return the value TRUE with a statement such as =RETURN(TRUE) if the solution process is to continue, or FALSE if the solution process should stop at this point.

Remarks

If a problem has not been completely defined, SOLVER.SOLVE returns the #N/A error value. Otherwise, the Solver application is started and the problem specifications are passed to it. When the solution process is complete, SOLVER.SOLVE returns an integer value indicating the stopping condition:

Value	Stopping condition
0	Solver found a solution. All constraints and optimality conditions are satisfied.
1	Solver has converged to the current solution. All constraints are satisfied.

2	Solver cannot improve the current solution. All constraints are satisfied.
3	Stop chosen when the maximum iteration limit was reached.
4	The Set Cells values do not converge.
5	Solver could not find a feasible solution.
6	Solver stopped at user's request.
7	The conditions for Assume Linear Model are not satisfied.
8	The problem is too large for Solver to solve.
9	Solver encountered an error value in a target or constraint cell.
10	Stop chosen when the maximum time limit was reached.
11	There is not enough memory available to solve the problem.
12	Another Excel instance is using SOLVER.DLL. Try again later.
13	Error in model. Please verify that all cells and constraints are valid.

Related Function

SOLVER.FINISH   Equivalent to clicking OK in the Solver Results dialog box that appears when the solution process is complete

SORT

Equivalent to clicking the Sort command on the Data menu. Sorts the rows or columns of the selection according to the contents of a key row or column within the selection. Use SORT to rearrange information into ascending or descending order.

Syntax 1

For Worksheet and macro sheets

SORT(orientation, key1, order1, key2, order2, key3, order3, header, custom, case)

SORT?(orientation, key1, order1, key2, order2, key3, order3, header, custom, case)

Syntax 2

For PivotTable reports

SORT(orientation, key1, order1, type, custom)

SORT?(orientation, key1, order1, type, custom)

Orientation is a number specifying whether to sort by rows or columns. Enter 1 to sort top to bottom or 2 to sort left to right.

Key1 is a reference to the cell or cells you want to use as the first sort key. The sort key identifies which column to sort by when sorting rows or which row to sort by when sorting columns. For a PivotTable report, if type is 1, then key1 is a cell reference which indicates what value to sort by. There are two ways to specify sort keys:

Type of key	Examples
An R1C1-style reference in the form of text. If the reference is relative, it is assumed to be relative to the active cell in the selection.	"C2" or "C[1]" or "Price"

Order1 specifies whether to sort the row or column containing key1 in ascending or descending order. Enter 1 to sort in ascending order or 2 to sort in descending order.

Key2, order2, key3, and order3 are similar to key1 and order1. Key2 specifies the second sort key, and order2 specifies whether to sort the row or column containing key2 in ascending or descending order. Key3 and order3 work similarly.

Header is a number indicating how Microsoft Excel is to handle headers on list.

Header	Defined
0	Microsoft Excel will guess if there is a header
1	Forces Microsoft Excel to assume there is a header
2 or omitted	Forces Microsoft Excel to assume there is no header

Type is a number specifying whether to sort the field by labels or values. Use one to sort by values or two to sort by labels.

Custom is a number that specifies what kind of custom sorting you want. This corresponds to the First Key Sort Order drop-down box in the Sort Options dialog box. For a PivotTable report, custom is a number indicating what custom sort order to use when sorting labels.

Number	Type of sort
1	Normal
2	Weekdays in abbreviated form ("Sun", "Mon", and so on)
3	Weekdays
4	Months in abbreviated form ("Jan" "Feb", and so on)

5	Months

Case is a logical value that determines whether the sort is case sensitive. If TRUE, the sort is case sensitive. If FALSE or omitted, the sort will not be case sensitive.

Tip If you want to sort using more than three keys, then sort the data three keys at a time, starting with the least important group of keys and progressing to the most important group, but listing the most important key first within each group.

Remarks

In the dialog box form of this function, if the header argument is omitted, then Microsoft Excel will guess whether or not there are headers.

SOUND.NOTE

This function should not be used in Microsoft Excel 97 or later because sound notes are available only in Microsoft Excel 95 or earlier versions.

Records sound into or erases sound from a cell note or imports sound from another file into a cell note. This function requires that you have recording hardware installed in your computer, and you must be running Microsoft Windows version 3.1 or later, or Apple system software version 6.07 or later.

Syntax 1

Recording or erasing sound

SOUND.NOTE(cell_ref, erase_snd)

Syntax 2

Importing sound from another file

SOUND.NOTE(cell_ref, file_text, resource)

Cell_ref is a reference to the cell containing a note into which you want to record or import sounds or from which you want to erase a sound.

Erase_snd is a logical value specifying whether to erase the sound in the note. If erase_snd is TRUE, Microsoft Excel erases only the sound from the note. If FALSE or omitted, Microsoft Excel displays the Record dialog box so that you can record sound into the note.

File_text is the name of a file containing sounds.

Resource is the number or name of a sound resource in file_text that you want to import into your note.

- This argument applies only to Microsoft Excel for the Macintosh.
- If resource is omitted, Microsoft Excel uses the first resource in the file.
- If the file does not contain a sound resource with the specified name or number, Microsoft Excel halts the macro and displays an error message.

Remarks

- To find out if a cell has sound attached to it, use GET.CELL(47).

- Sounds notes are not available in Microsoft Excel 97 or later.

Examples

The following macro formula erases the sound, if present, from cell A1 on the active sheet:

```
SOUND.NOTE(!$A$1, TRUE)
```

The following macro formula displays the Record dialog box so that you can record sound into a note for cell A1 on the active sheet:

```
SOUND.NOTE(!$A$1)
```

In Microsoft Excel for Windows, the following macro formula imports the sound from a file named CHIMES.WAV into a note for the cell named Doorbell on the active sheet:

```
SOUND.NOTE(!Doorbell, "C:\SOUNDS\CHIMES.WAV")
```

In Microsoft Excel for the Macintosh, the following macro formula imports a sound called Chimes from a file named SOFT SOUNDS into a note for the cell named Doorbell on the active sheet:

```
SOUND.NOTE(!Doorbell, "HARD DISK:SOUNDS:SOFT SOUNDS", "Chimes")
```

Related Functions

NOTE   Creates or changes a cell note

SOUND.PLAY   Plays the sound from a cell note or a file

SOUND.PLAY

This function should not be used in Microsoft Excel 97 or later because sound notes are available only in Microsoft Excel 95 or earlier versions.

Plays the sound from a cell note or a file. Equivalent to clicking the Note command on the Insert menu and clicking the Play button, or clicking the Note command on the Insert menu, clicking the Import button, and then opening a file, selecting a sound, and clicking the Play button. To play sounds in Microsoft Excel for Windows, you must have a sound board installed in your computer.

Syntax

SOUND.PLAY(cell_ref, file_text, resource)

Cell_ref   is a reference to the cell note containing sound that you want to play. If cell_ref is omitted, Microsoft Excel plays the sound from the active cell, or from a file if you specify one.

File_text   is the name of a file containing sounds. If cell_ref is specified, file_text is ignored.

Resource   is a number or name given as text specifying a sound resource in file_text that you want to play.

- This argument applies only to Microsoft Excel for the Macintosh.
- If cell_ref is specified, resource is ignored.

- If resource is omitted, Microsoft Excel uses the first sound resource in the file.
- If the file does not contain a sound resource with the specified name or number, Microsoft Excel halts the macro and displays an error message.

Related Function

SOUND.NOTE Records or imports sound into or erases sound from cell notes

SPELLING

Equivalent to clicking the Spelling command on the Tools menu. Checks the spelling of words in the current selection.

Syntax

SPELLING(custom_dic, ignore_uppercase, always_suggest)

Custom_dic is the filename of the custom dictionary to examine if words are not found in the main dictionary. If custom_dic is omitted, the currently specified dictionary is used.

Ignore_uppercase is a logical value corresponding to the Ignore UPPERCASE check box.

If ignore_uppercase is	Microsoft Excel will
TRUE	Ignore words in all uppercase letters
FALSE	Check words in all uppercase letters
Omitted	Use the current setting

Always_suggest is a logical value corresponding to the Always Suggest check box.

If always_suggest is	Microsoft Excel will
TRUE	Display a list of suggested alternate spellings when an incorrect spelling is found
FALSE	Wait for user to input the correct spelling
Omitted	Use the current setting

Related Function

SPELLING.CHECK Checks the spelling of a word

SPELLING.CHECK

Checks the spelling of a word. Returns TRUE if the word is spelled correctly; FALSE otherwise.

Syntax

SPELLING.CHECK(**word_text**, custom_dic, ignore_uppercase)

Word_text is the word whose spelling you want to check. It can be text or a reference to text.

Custom_dic is the filename of a custom dictionary to examine if the word is not found in the main dictionary.

Ignore_uppercase is a logical value corresponding to the Ignore Words In Uppercase check box. If ignore_uppercase is TRUE, the check box is selected, and Microsoft Excel ignores words in all uppercase letters; if FALSE, the check box is cleared, and Microsoft Excel checks all words; if omitted, the current setting is used.

Remarks

This function does not have a dialog-box form. To display the Spelling dialog box, use SPELLING.

Related Function

SPELLING	Checks the spelling of words in the current selection

SPLIT

Equivalent to choosing the Split command from the Window menu or to dragging the split bar in the active window's scroll bar. Splits the active window into panes. Use SPLIT when you want to view different parts of the active sheet at the same time.

Syntax

SPLIT(col_split, row_split)

Col_split specifies where to split the window vertically and is measured in columns from the left of the window.

Row_split specifies where to split the window horizontally and is measured in rows from the top of the window.

If an argument is 0 and there is a split in that direction, the split is removed. If an argument is omitted, a split in that direction is not changed.

Related Function

FREEZE.PANES	Freezes or unfreezes the panes of a window

SQL.BIND

Specifies where results from a SQL query are placed when they are retrieved with SQL.RETRIEVE. If this function is not available, you must install the Microsoft ODBC add-in (XLODBC.XLA).

Syntax

SQL.BIND(**connection_num**, column, reference)

Connection_num is the unique connection ID of the data source for which you want to define storage.

- Connection_num was returned by a previously executed SQL.OPEN function.
- If connection_num is not valid, then SQL.BIND returns the #VALUE! error value.

Column is the number of the result column that is to be bound. Result columns are numbered from left to right starting with 1. If column is omitted then all bindings for connection_num are removed. Column number 0 contains row numbers for the result set. If column number 0 is bound then SQL.RETRIEVE will return row numbers in the bound location.

Reference is a single cell reference on the worksheet where the results should be placed. If reference is omitted, the binding is removed for the column.

When SQL.RETRIEVE is called, the result rows in this column will be placed in the reference cell and the cells immediately below reference. The number of rows that will be retrieved is one of the SQL.RETRIEVE arguments.

Remarks

- If SQL.BIND is completed successfully then it will return a vertical array listing the bound columns on the current connection. If SQL.BIND is unable to bind the result column then it will return the error value #N/A. In such a case SQL.BIND will place error information in memory for the SQL.ERROR function, if such information is available.
- SQL.BIND tells the ODBC interface where to place results when they are retrieved using SQL.RETRIEVE. Binding is not necessary but can be useful if you want the results from different columns to be placed in disjoint worksheet locations.
- If bindings are used, SQL.BIND must be called once for each column in the result set. If a result column is not bound then it will not be returned. A binding remains valid for as long as connection_num is open.
- Call SQL.BIND after calling SQL.OPEN and SQL.EXEC.QUERY, but before calling SQL.RETRIEVE or SQL.RETRIEVE.TO.FILE. Calls to SQL.BIND will not affect results that have already been retrieved.

Example

`SQL.BIND(conn1,1,"[Book1]Sheet1!C1")` stores data obtained from the data source conn1 on Sheet1 from left to right in cell C1, starting with column1.

Related Functions

SQL.OPEN	Establishes a connection with a data source
SQL.EXEC.QUERY	Sends a query to a data source
SQL.RETRIEVE.TO.FILE	Retrieves query results and places them in a file
SQL.RETRIEVE	Retrieves query results
SQL.GET.SCHEMA	Gets information about a connected data source.
SQL.CLOSE	Closes a connection to a data source.
SQL.ERROR	Returns detailed error information

SQL.CLOSE

Terminates a connection to an external data source. If this function is not available, you must install the Microsoft ODBC add-in (XLODBC.XLA).

Syntax

SQL.CLOSE(**connection_num**)

Connection_num is the unique connection ID of the data source from which you wish to disconnect.

- Connection_num is returned by a previously executed SQL.OPEN function.
- If connection_num is not valid, SQL.CLOSE returns the #VALUE! error value.

Remarks

- If the connection is successfully terminated SQL.CLOSE will return zero and the connection ID number is then no longer valid.
- If SQL.CLOSE is unable to disconnect with the data source then it will return the error value the #N/A error value. In such a case SQL.CLOSE will place error information in memory for the SQL.ERROR function, if such information is available.
- SQL.CLOSE works with data sources in much the same manner as FCLOSE works with files. If the call is successful then SQL.CLOSE will terminate the specified data source connection.

Example

SQL.CLOSE`(conn1)` will close the connection with connection_num conn1

Related Functions

SQL.OPEN Establishes a connection with a data source

SQL.EXEC.QUERY Sends a query to a data source

SQL.BIND Specifies storage for a result column

SQL.RETRIEVE.TO.FILE Retrieves query results and places them in a file

SQL.RETRIEVE Retrieves query results

SQL.GET.SCHEMA Gets information about a connected data source.

SQL.ERROR Returns detailed error information

SQL.ERROR

Returns detailed error information when it is called after a previous XLODBC.XLA function call has failed. If this function is not available, you must install the Microsoft ODBC add-in (XLODBC.XLA).

Syntax

SQL.ERROR()

Calling SQL.ERROR returns detailed error information in a two dimensional array. Each row in the array describes exactly one error. If a function call generates multiple errors, a row will be created for each error. When SQL.ERROR is processed successfully, all SQL.ERROR information is cleared. Also, all SQL.ERROR information is automatically removed whenever an ODBC function completes successfully.

Each row will have exactly three fields. The information in these three fields is obtained through the SQLERROR API function call. These fields are:

- A textual message describing the error.
- The ODBC error class and subclass as a character string.
- The data source native error code as a numeric value.

If one or more of these fields is not available for the type of error that was encountered, the field will be left blank. For more information on the meaning of these three fields, refer to Chapter 24, "ODBC Function Reference", in the Microsoft Open Database Connectivity Programmer's Reference for the SQLError API function. See also Appendix A, "ODBC Error Codes" in the same manual.

Remarks

- SQL.ERROR cannot provide information on Excel errors.
- If no error information is available when SQL.ERROR is called, then it well return the error value #N/A but does not post any error information to SQL.ERROR.
- SQL.ERROR stores and returns error information by processing SQL.ERROR (in the ODBC API reference) in a loop until SQL_NO_DATA_FOUND is encountered. In the SQL.ERROR function, the error information is automatically defined and stored in memory whenever an XLODBC.XLA function fails. If the call is successful then SQL.ERROR will return the error information available. If SQL.ERROR fails, it will return the error value #N/A.

Example

When entered as an array formula, the following example will return error information about each argument in, for example, SQL.GET.QUERY.

```
SQL.ERROR()
```

Related Functions

SQL.OPEN	Establishes a connection with a data source
SQL.EXEC.QUERY	Sends a query to a data source
SQL.BIND	Specifies storage for a result column
SQL.RETRIEVE.TO.FILE	Retrieves query results and places them in a file
SQL.RETRIEVE	Retrieves query results
SQL.CLOSE	Closes a data source connection

SQL.EXEC.QUERY

Sends a query to a data source using an existing connection. If this function is not available, you must install the Microsoft ODBC add-in (XLODBC.XLA).

Syntax

SQL.EXEC.QUERY(**connection_num**, **query_text**)

Connection_num is the unique connection ID of the data source you want to query.

- Connection_num is returned by a previously executed SQL.OPEN function.
- If connection_num is not valid, SQL.EXEC.QUERY returns the #VALUE! error value.

Query_text is the SQL language query that is to be executed on the data source. The query must follow the SQL syntax guidelines in the Appendix of the Microsoft Excel ODBC Developers Guide.

- If SQL.EXEC.QUERY is unable to execute query_text on the specified data source, SQL.EXEC.QUERY returns the #N/A error value.
- Excel limits strings to a length of 255 characters. If query_text needs to be longer than 255 characters then query_text should be a vertical array or vertical range of cells. The values in the array will be joined together to form the complete SQL query.

Remarks

- Before calling SQL.EXEC.QUERY a connection must be established with a data source using SQL.OPEN. A successful call to SQL.OPEN returns a unique connection ID number. SQL.EXEC.QUERY uses that connection ID number to send SQL language queries to the data source.
- Any results generated from the query will not be returned immediately--SQL.EXEC.QUERY only executes the query. Retrieving results is handled by the functions SQL.RETRIEVE and SQL.RETRIEVE.TO.FILE.
- If SQL.EXEC.QUERY is called using a previously used connection ID number, all pending results on that connection will automatically be discarded. The connection ID will then refer to the new query and its results.
- If SQL.EXEC.QUERY is unable to successfully execute the query on the specified data source then an error value will be returned. In such a case SQL.EXEC.QUERY will place error information in memory for the SQL.ERROR function, if such information is available. If SQL.EXEC.QUERY is able to successfully execute the query on the specified connection it will return one of three values depending on the type of SQL statement that was executed.
    - If it was a SELECT statement then SQL.EXEC.QUERY will return the number of result columns available.
    - If it was an UPDATE, INSERT, or DELETE statement then SQL.EXEC.QUERY will return the number of rows affected by the statement.

- If it was a legal SQL query that is not in one of the categories above, SQL.EXEC.QUERY will return 0 (zero).

Example

`SQL.EXEC.QUERY(conn1, "SELECT Custmr_ID, Due_Date FROM Orders WHERE Order_Amt > 100")` executes a SQL query from a SQL table named "Orders"

Related Functions

SQL.OPEN	Establishes a connection with a data source
SQL.BIND	Specifies storage for a result column
SQL.RETRIEVE.TO.FILE	Retrieves query results and places them in a file
SQL.RETRIEVE	Retrieves query results
SQL.GET.SCHEMA	Gets information about a connected data source.
SQL.CLOSE	Closes a data source connection
SQL.ERROR	Returns detailed error information

SQL.GET.SCHEMA

Returns information about the structure of the data source on a particular connection. The return value from a successful call to SQL.GET.SCHEMA depends on the type of information that was requested. A list of the accepted requests and their respective return values is listed in the syntax section below.

If this function is not available, you must install the Microsoft ODBC add-in (XLODBC.XLA).

Syntax

SQL.GET.SCHEMA(**connection_num**, **type_num**, qualifier_text)

Connection_num is the unique connection ID of the data source you want information about.

- Connection_num is returned by a previously executed SQL.OPEN function.
- If connection_num is not valid, SQL.GET.SCHEMA returns the #VALUE! error value.

Type_num specifies the type of information you want returned. The following is a list of valid type_num values.

Type_num	Returns
1	A list of available data sources, as a vertical array.
2	A list of databases on the current connection, as a vertical array .
3	A list of owners in a database on the current connection, as a vertical array.

4	A list of tables for a given owner and database on the current connection, as a vertical array.
5	A list of columns in a particular table and their data types, as a two-dimensional array. The returned array will have two fields and will have a row for each column in the table. The first field will be the name of the column. The second field is the data type of the column. The data type will be a number that corresponds to the ODBC C header file data types. These #define numbers are found in Microsoft Excel ODBC Developer Guide.
6	User ID of the current user
7	Name of the current database.
8	The name of the data source as given in the ODBC.INI file.
9	The name of the data source DBMS (i.e. Oracle, SQL Server, etc.).
10	The server name for the data source.
11	The terminology used by the data source to refer to owners (i.e. "owner", "Authorization ID", "Schema", etc.).
12	The terminology used by the data source to refer to tables (i.e. "table", "file", etc.).
13	The terminology used by the data source to refer to qualifiers (i.e. "database" or "directory").
14	The terminology used by the data source to refer to procedures (i.e. "database procedure", "stored procedure", or "procedure").

Qualifier_text is only included for type_num values of 3, 4 and 5. It is a text string used to qualify the search for the requested information and should be enclosed by quotation marks.

Type_num	**Qualifier_text**
3	The value of qualifier_text should be the name of a database in the current data source. SQL.GET.SCHEMA will then only return the names of table owners in that database.

4	The value of qualifier_text should be both a database name and an owner name. The syntax of qualifier_text is "DatabaseName.OwnerName". A period is used to separate the two names. SQL.GET.SCHEMA will then return an array of table names that are located in the given database and owned by the given owner.
5	The value of qualifier_text should be the name of a table. Information about the columns in that table will be returned.

Remarks

- If SQL.GET.SCHEMA is unable to find the requested information then it will return the error value #N/A. In such a case SQL.GET.SCHEMA will place error information in memory for the SQL.ERROR function, if such information is available.
- SQL.GET.SCHEMA works with the ODBC functions SQLGetInfo and SQLTables to find the requested information. Refer to the Microsoft Excel ODBC Programmer Guide for more information on these two functions.

Example

`SQL.GET.SCHEMA(conn1,7)` returns the name of the current database.

`SQL.GET.SCHEMA(conn1,9)` returns the name of the DBMS used by the data source.

Related Functions

SQL.OPEN	Establishes a connection with a data source
SQL.EXEC.QUERY	Sends a query to a data source
SQL.BIND	Specifies storage for a result column
SQL.RETRIEVE.TO.FILE	Retrieves query results and places them in a file
SQL.RETRIEVE	Retrieves query results
SQL.CLOSE	Closes a data source connection
SQL.ERROR	Returns detailed error information

SQL.OPEN

Establishes a connection with a data source. If the connection is successfully established SQL.OPEN will return a connection ID number. Use the connection ID number with other ODBC macro functions to identify a connection.

If this function is not available, you must install the Microsoft ODBC add-in (XLODBC.XLA).

Syntax

SQL.OPEN(connection_string, output_ref, driver_prompt)

Connection_string is a text string that contains the information necessary to establish a connection to a data source. Any data-source-name that is used in connection_string must be an existing data source name defined with ODBC Setup or the ODBC Administration Utility.

- Connection_string must follow the format described in Chapter 24, "ODBC Function Reference", of the Microsoft Open Database Connectivity Programmer's Reference for SQLDriverConnect. In this string the user supplies the data source name, one or more user ID's, one or more passwords, and any other information necessary to successfully connect to a DBMS. An example of a SQL.OPEN connection_string entered would be: "DSN=MyServer; UID=dbayer; PWD=123; Database=pubs"
- Enter the connection_string as an array when the length exceeds 255 characters. Or enter connection_string as an array of cells containing the same information. The connection string should be horizontal array.

Output_ref is a single cell reference where you want the completed connection string to be placed. Use output_ref when you want SQL.OPEN to return the completed connection string. If output_ref is omitted, a completed connection string will not be returned.

Driver_prompt is a number from 1 to 4 specifying if and how you want to be prompted by the driver. This sets the fDriverCompletion flag in ODBC's SQLDriverConnect.

Number	Description
1	Always brings up a dialog box.
2	Bring up dialog only if there is not enough information to connect. The driver uses information from the connection string and from the data source specification as defaults.
3	Same as 2, but the driver grays and disables any prompts for information not needed.
4	If the connection string is unsuccessful, do not bring up a dialog box.

Remarks

- If SQL.OPEN is unable to connect with the information provided then it will return the error value #N/A. In such a case, SQL.OPEN will place error information in memory for the SQL.ERROR function, if more information is available.
- If the call is successful then SQL.OPEN will return a unique connection ID number that can be used in future function calls to identify the connection.
- If connection_array does not allow SQL.OPEN to connect to a data source, then the error value #N/A will be returned.

Example

`conn1=SQL.OPEN('DSN=NWind;DBQ=C:\MSQUERY;FIL=dBASE4',C15, 2)` sets the name conn1 to the return value of SQL.OPEN, which connects to the NWind data source, specifies where to place the connection string, and displays the driver dialog box only if additional information is needed.

Related Functions

SQL.EXEC.QUERY Sends a query to a data source

SQL.BIND   Specifies storage for a result column

SQL.RETRIEVE.TO.FILE   Retrieves query results and places them in a file

SQL.RETRIEVE   Retrieves query results

SQL.GET.SCHEMA   Gets information about a connected data source.

SQL.CLOSE   Closes a data source connection

SQL.ERROR   Returns detailed error information

SQL.RETRIEVE

Retrieves all or part of the results from a previously executed query. The connection used must have already been established using the macro function SQL.OPEN. Also, a query must already have been executed using SQL.EXEC.QUERY and results must be pending.

If this function is not available, you must install the Microsoft ODBC add-in (XLODBC.XLA).

Syntax

SQL.RETRIEVE(**connection_num**, destination_ref, max_columns, max_rows, col_names_logical, row_nums_logical, named_rng_logical, fetch_first_logical)

Connection_num   is the unique connection ID for a data source. The data source specified must have pending query results. Pending query results are generated by a call to SQL.EXEC.QUERY on the same connection.

- If there are no pending results on the connection SQL.RETRIEVE returns the #N/A error value.
- If connection_num is not valid, SQL.EXEC.QUERY returns the #VALUE! error value.

Destination_ref   specifies where the results should be placed. It is either a reference to a single cell or it is omitted.

- If destination_ref refers to a single cell then SQL.RETRIEVE will return all of the pending results in the cells to the right, below, and including destination_ref. This is the same convention used in Microsoft Excel when multiple cells are pasted into a single-cell selection. Any previous values contained in the destination cells will be overwritten without confirmation.
- If destination_ref is omitted then the bindings established by previous calls to SQL.BIND will be used to return results. If no such bindings exist for the current connection then SQL.RETRIEVE will return the #REF! error value. If a particular result column has not been bound then its results will be discarded. Max_rows specifies the number of rows that will be returned under each bound column. The first row of results will be placed in the bound cell and any additional rows will be placed in the rows immediately under the bound cell.

Max_columns   is the maximum number of columns to be retrieved. It is only used when destination_ref is not omitted.

- If max_columns specifies more columns than are available in the results, SQL.RETRIEVE will place data in the columns for which data is available and clear the additional columns.
- If max_columns specifies fewer columns than are available in the results, the rightmost result columns will be discarded to fit the chosen size. Column position will be determined by the order in which the data source returned them.
- If max_columns is omitted then all of the result columns will be returned.

Max_rows is the maximum number of rows to be returned.

- If max_rows specifies more rows than are available in the results, SQL.RETRIEVE will place data in the rows for which data is available and clear the additional rows.
- If max_rows specifies fewer rows than are available in the results, SQL.RETRIEVE will place data in the selected rows but will not discard the additional rows. These extra rows can be retrieved via additional calls to SQL.RETRIEVE. This process is described in the fetch_first_logical argument description.
- If max_rows is omitted then all rows in the result set will be returned.

Col_names_logical is a logical value which, if TRUE, causes the column names to be returned as the first row of results. It FALSE or omitted, the column names will not be returned.

Row_nums_logical is used only when destination_ref is included. If row_nums_logical is TRUE then the first column in the result set will contain row numbers. If FALSE then row numbers will not be returned. This column of row numbers will not have a column name and the column heading will be left blank. Row numbers can also be retrieved by binding column number 0 with SQL.BIND.

Named_rng_logical is a logical value which, if TRUE, sets each column of results to be declared as a named range on the worksheet. The name of the each range will be the result column name. The named range will only include the rows that were fetched with this SQL.RETRIEVE function call. If FALSE, the results will not be declared as a named range.

Fetch_first_logical is a logical value that allows you to request results from the beginning of the result set.

- If the first call to SQL.RETRIEVE did not return all of the rows in the result set then SQL.RETRIEVE may be called again to return the next set of rows. This process can be repeated until no more result rows are available, at which time SQL.RETRIEVE will return the value 0 (zero). This will not halt the running of the macro. During each of these calls, including the first call, fetch_first_logical should be set to FALSE.
- If you want to move the cursor back to the beginning of the result set then fetch_first_logical should be set to TRUE. This causes the same SQL query text to be executed again on the data source. The cursor will then be positioned at the top of the result set and SQL.RETRIEVE will fill destination_ref beginning with the first row of results. Further calls to SQL.RETRIEVE, for the purpose of retrieving additional rows, can then be made with fetch_first_logical set to FALSE .

Remarks

- Before calling SQL.RETRIEVE a connection must be established with a data source using SQL.OPEN.
- If SQL.RETRIEVE is unable to retrieve the results on the specified data source then an error value will be returned. In such a case SQL.RETRIEVE will place error information in memory for the SQL.ERROR function, if such information is available.
- If SQL.RETRIEVE is able to successfully return rows of results on the specified connection it will return the number of rows that were actually returned. If there were no results pending on the connection then SQL.RETRIEVE will return the #N/A error value.If no data was found then SQL.RETRIEVE returns 0 (zero).
- A successful call to SQL.OPEN returns a unique connection ID number, which is used in a call to SQL.EXEC.QUERY to send a SQL language query. Following this call to SQL.EXEC.QUERY, SQL.RETRIEVE uses the same connection ID number to retrieve query results.

Example

`SQL.RETRIEVE(conn1,sheet1!C1,1)` stores data obtained from the data source conn1 on Sheet1 from left to right in cell C1, using only column 1.

Related Functions

SQL.OPEN	Establishes a connection with a data source
SQL.EXEC.QUERY	Sends a query to a data source
SQL.BIND	Specifies storage for a result column
SQL.RETRIEVE.TO.FILE	Retrieves query results and places them in a file
SQL.GET.SCHEMA	Gets information about a connected data source.
SQL.CLOSE	Close a data source connection
SQL.ERROR	Returns detailed error information

SQL.RETRIEVE.TO.FILE

Retrieves all of the results from a previously executed query and places them in a file. The connection used must have already been established using the macro function SQL.OPEN. Also, a query must already have been executed using SQL.EXEC.QUERY and results must be pending.

If this function is not available, you must install the Microsoft ODBC add-in (XLODBC.XLA).

Syntax

SQL.RETRIEVE.TO.FILE(**connection_num**, **destination**, col_names_logical, column_delimiter)

Connection_num is the unique connection ID for a data source. The data source specified must have query results pending. Pending results were generated by a previous call to SQL.EXEC.QUERY on the same connection.

- If there are no pending results on the connection SQL.RETRIEVE.TO.FILE returns the #N/A error value. The file is not affected.
- If connection_num is not valid, SQL.RETRIEVE.TO.FILE returns the #VALUE! error value.

Destination specifies the name and path of the file where the results should be placed. SQL.RETRIEVE.TO.FILE will open the specified file and fill it with the entire result set.

- The format of the data in the file will be compatible with the Microsoft Excel ".CSV" format. The overall format will be that columns will be separated by the value in column_delimiter (see below) and the individual rows will be separated by a linefeed/carriage-return.
- If the file specified by destination cannot be opened then the error value #N/A will be returned by SQL.RETRIEVE.TO.FILE.
- If the file already exists its previous contents will be overwritten by SQL.RETRIEVE.TO.FILE.

Col_names_logical is a logical value that, if TRUE, allows the column names to be returned as the first row of data. If FALSE or omitted, the column names will not be returned.

Column_delimiter is the value that will be used to separate the elements in each row. If column_delimiter is omitted then a tab will be used. If another value is desired then it should be enclosed in quotation marks. Possible values for column_delimiter might be: "," or ";" or " ". The string "tab" can also be used to specify a tab separator (even though this is redundant, since a tab is the default).

Remarks

- If SQL.RETRIEVE.TO.FILE is unable to retrieve the results on the specified connection then an error value will be returned. In such a case SQL.RETRIEVE.TO.FILE will place error information in memory for the SQL.ERROR function, if such information is available.
- If SQL.RETRIEVE.TO.FILE is able to successfully return rows of results on the specified connection and place them in a file it will return the number of rows that were actually written to the file. If there were no results pending on the connection then SQL.RETRIEVE.TO.FILE will return the #N/A error value and the file will not be created or modified.
- Before calling SQL.RETRIEVE.TO.FILE a connection must be established with a data source using SQL.OPEN.
- A successful call to SQL.OPEN returns a unique connection ID number, which can be used in a call to SQL.EXEC.QUERY to send a SQL language query. Following this call to SQL.EXEC.QUERY, SQL.RETRIEVE.TO.FILE uses the same connection ID number to retrieve query results and place them in a file.

Example

SQL.RETRIEVE.TO.FILE(conn1,"C:\MSQUERY\RESULTS1.QRY",TRUE,",") retrieves the results of a previously executed query and places them in the file RESULTS1.QRY, with column names that are comma delimited.

Related Functions

SQL.OPEN	Establishes a connection with a data source
SQL.EXEC.QUERY	Sends a query to a data source
SQL.BIND	Specifies storage for a result column
SQL.RETRIEVE	Retrieves query results
SQL.GET.SCHEMA	Gets information about a connected data source.
SQL.CLOSE	Closes a data source connection
SQL.ERROR	Returns detailed error information

STANDARD.FONT

Sets the attributes of the standard font in Microsoft Excel version 2.2 and earlier. This function is included only for macro compatibility. To define and apply a style in Microsoft Excel version 5.0 or later, use the APPLY.STYLE and DEFINE.STYLE functions.

Syntax

STANDARD.FONT(name_text, size_num, bold, italic, underline, strike, color, outline, shadow)

The arguments for this function are the same as those for FORMAT.FONT.

Related Functions

APPLY.STYLE	Applies a style to the selection
DEFINE.STYLE	Defines a cell style
FORMAT.FONT	Applies a font to the selection

STANDARD.WIDTH

Sets the default width used for all columns that you have not previously adjusted on the active worksheet.

Syntax

STANDARD.WIDTH(standard_num)

Standard_num specifies how wide you want the columns to be in units of one character of the font corresponding to the Normal cell style.

STEP

Stops the normal flow of a macro and calculates it one cell at a time. Running a macro one cell at a time is called single-stepping and is very useful when you are debugging a macro. Use the STEP function, instead of clicking the Step Into button in the Macro dialog box when you want to start single-stepping at a specific line in a macro. The Macro dialog box appears when you click the Macros command (Tools menu, Macro submenu).

Syntax

STEP( )

Remarks

- When Microsoft Excel encounters a STEP function, it stops running the macro and displays a dialog box. The dialog box tells you which cell in the macro Microsoft Excel is about to calculate, and what formula is in that cell. You can click Step to carry out the next instruction; click Evaluate to calculate part of the formula; click Halt to interrupt the macro; or click Continue to continue the macro without single-stepping.
- When placed at the beginning of a macro, STEP is equivalent to clicking the Macro command on the Tools menu and clicking the Step Into button in the Macro dialog box.
- To step through the calculation of a custom function, place the STEP function at the start of the custom function.

Related Functions

HALT   Stops all macros from running

RUN   Runs a macro

STYLE

Checks the fonts for a bold and/or italic font and applies it to the current selection in Microsoft Excel for the Macintosh version 1.5 or earlier. If no appropriate font is available, Microsoft Excel finds the most similar font available and formats the selection using that font. This function is included only for macro compatibility. If you want to change a font to bold or italic, use the FONT.PROPERTIES function.

Syntax

STYLE(bold,italic)

STYLE?(bold,italic)

Related Function

FONT.PROPERTIES   Applies a font to the selection

SUBSCRIBE.TO

Inserts the contents of the edition into the active sheet at the point of the current selection. Use SUBSCRIBE.TO to incorporate editions published from other workbooks into your Microsoft Excel worksheets and macro sheets. SUBSCRIBE.TO returns TRUE if successful.

Syntax

SUBSCRIBE.TO(**file_text, format_num**)

Important   This function is only available if you are using Microsoft Excel for the Macintosh with system software version 7.0 or later.

File_text   is the name, as a text string, of the edition you want to insert into the active sheet. Unless file_text is in the current folder, supply the full path of the workbook. If file_text cannot be found, SUBSCRIBE.TO returns the #VALUE! error value and interrupts the macro.

Remarks

- If a single cell is selected, the data from the edition file is placed into as large a range of cells as is required by the data. Data already present in those cells is replaced. If the data is a picture, it is inserted from the upper-left corner of the selected cell.
- If a range of cells is selected, and the range is not big enough to contain the edition data, Microsoft Excel displays a dialog box asking if you want to clip the data to fit the range.

Format_num is the number 1 or 2 and specifies the format type of the file you are subscribing to.

Format_num	Format type
1 or omitted	Picture
2	Text (includes BIFF, VALU, TEXT, and CSV formats)

Related Functions

CREATE.PUBLISHER Creates a publisher from the selection

EDITION.OPTIONS Sets publisher and subscriber options

GET.LINK.INFO Returns information about a link

SUBTOTAL.CREATE

Equivalent to clicking the Subtotals command on the Data menu. Generates a subtotal in a list or database.

Syntax

SUBTOTAL.CREATE(**at_change_in**, **function_num**, **total**, replace, pagebreaks, summary_below)

SUBTOTAL.CREATE?(at_change_in, function_num, total, replace, pagebreaks, summary_below)

At_change_in is a column offset corresponding to the At Each Change In text box on the Subtotal dialog box.

Function_Num is a number corresponding to the Use Function list box specifying which function you want to use in subtotaling your data.

Function	Function_Num
SUM	1

COUNTA	2
AVERAGE	3
MAX	4
MIN	5
PRODUCT	6
COUNT	7
STDEV	8
STDEVP	9
VAR	10
VARP	11

Total is an array of column offsets corresponding to the Add Subtotal To list box. Indicates which columns you want aggregated according to function_num; for example, {4,5}

Replace is a logical value which, if TRUE, causes any previous subtotals to be replaced by new subtotals. If FALSE or omitted, subtotals will not be replaced.

PageBreaks is a logical value corresponding to the Page Break Between Groups check box which, if TRUE, creates a page break below each subtotal. If FALSE or omitted, does not create a page break below each subtotal.

Summary_Below is a logical value corresponding to the Summary Below Data check box which, if TRUE, places subtotal rows below the data they refer to, and a grand total at the bottom of the list. If FALSE, places subtotal rows above the data they refer to, and a grand total just below the header.

Related Function

SUBTOTAL.REMOVE Removes all previously existing subtotals and grand totals in a list

SUBTOTAL.REMOVE

Equivalent to clicking the Subtotal command on the Data menu, and then clicking the Remove All button in the Subtotal dialog box. Removes all previously existing subtotals and grand totals in a list. Any page breaks and outlines will also be removed.

Syntax

SUBTOTAL.REMOVE()

Related Function

SUBTOTAL.CREATE   Generates a subtotal in a list or database

SUMMARY.INFO

Equivalent to clicking the Properties command on the File menu. Generates the summary information for the active workbook.

Syntax

SUMMARY.INFO(title, subject, author, keywords, comments)

SUMMARY.INFO?(title, subject, author, keywords, comments)

The arguments correspond to the text boxes on the Summary tab of the Properties dialog box. If any arguments are omitted, that text box is left empty.

Title   specifies a title for the file, not necessarily a file name. Long names can be entered, up to 255 characters.

Subject   is information pertaining to the subject matter of the workbook.

Author   is initially the name specified in the User Name box on the General tab in the Options dialog box, which appears from the Options command from the Tools menu. If this is omitted, the registered user of the copy of Microsoft Excel will be used.

Keywords   are keywords that can be later used in searching for the contents in the file.

Comments   is a comment that can be entered to help a user learn more about the contents or subject matter of the workbook.

Related Functions

FIND.FILE   Lets you search for files based on criteria such as author or creation date

GET.WORKBOOK   Returns information about a workbook sheet

TABLE

Equivalent to clicking the Table command on the Data menu. Creates a table based on the input values and formulas you define on a worksheet. Use data tables to perform a "what-if" analysis by changing certain constant values in your workbook to see how values in other cells are affected.

Syntax

TABLE(row_ref, column_ref)

TABLE?(row_ref, column_ref)

Row_ref   specifies the one cell to use as the row input for your table.

- Row_ref should be either an external reference to a single cell on the active worksheet, such as !$A$1 or !Price, or an R1C1-style reference to a single cell in the form of text, such as "R1C1", "R[-1]C[-1]", or "Price".

- If row_ref is an R1C1-style reference, it is assumed to be relative to the active cell in the selection.

Column_ref specifies the one cell to use as the column input for your table. Column_ref is subject to the same restrictions as row_ref.

TAB.ORDER

This function determines the order in which dialog controls will be selected when the user presses the TAB key.

Syntax

TAB.ORDER?( )

Remarks

- This function brings up the Tab Order dialog box and allows the user to select the order in which buttons will be selected when the TAB key is pressed.
- The BRING.TO.FRONT and SEND.TO.BACK macro functions can also be used to programmatically set up the tab order.

Related Functions

BRING.TO.FRONT	Puts the selected object or objects on top of all other objects
SEND.TO.BACK	Puts the selected object or objects behind all other objects

TERMINATE

Closes a dynamic data exchange (DDE) channel previously opened with the INITIATE function. Use TERMINATE to close a channel after you have finished communicating with another application.

Syntax

TERMINATE(**channel_num**)

Important Microsoft Excel for the Macintosh requires system software version 7.0 or later for this function.

Channel_num is the number returned by a previously run INITIATE function. Channel_num identifies a DDE channel to close.

If TERMINATE is not successful, it returns the #VALUE! error value.

Related Functions

EXEC	Starts another application
INITIATE	Opens a channel to another application

TEXT.BOX

Replaces characters in a text box or button with the text you specify.

Syntax

TEXT.BOX(**add_text**, object_id_text, start_num, num_chars)

Add_text   is the text you want to add to the text box or button.

Object_id_text   is the name of the text box or button to which you want to add text (for example, "Text 1" or "Button 2"). If object_id_text is omitted, it is assumed to be the selected item.

Start_num   is a number specifying the position of the first character you want to replace (or the position at which you want to insert characters if you do not want to replace any). If start_num is omitted, it is assumed to be 1.

Num_chars   is the number of characters you want to replace. If num_chars is 0, then no characters are replaced, and add_text is inserted starting at the position start_num. If num_chars is omitted, all the characters are replaced.

Examples

The following macro formula replaces the first five characters in a text box named "Text 5" with the text "Net Income":

```
TEXT.BOX("Net Income", "Text 5", 1, 5)
```

The following macro formula inserts the words "Account Summary for 1991" at the beginning of a text box named "Text 6":

```
TEXT.BOX("Account Summary for 1991", "Text 6", 1, 0)
```

Related Functions

CREATE.OBJECT	Creates an object
FONT.PROPERTIES	Applies a font to the selection
GET.OBJECT	Returns information about an object

TEXTREF

Converts text to an absolute reference in either A1- or R1C1-style. Use TEXTREF to convert references stored as text to references so that you can use them with other functions, such as OFFSET.

Syntax

TEXTREF(**text**, a1)

Text   is a reference in the form of text.

A1   is a logical value specifying the reference type of text. If a1 is TRUE, text is assumed to be an A1-style reference; if FALSE or omitted, text is assumed to be an R1C1-style reference.

Remarks

- If you use TEXTREF by itself in a cell, you will get the value contained in the cell specified by text, not the reference itself, because references are automatically converted into the contents of the referenced cell.

- If you use TEXTREF as a reference argument to a function, Microsoft Excel does not convert the reference to a value.

Tip You can convert a reference to text with REFTEXT, manipulate it with the REPLACE and MID functions, and convert it back to a reference with TEXTREF.

Examples

`TEXTREF("B7", TRUE)` equals the reference value $B$7

`TEXTREF("R5C5", FALSE)` equals the reference value R5C5

`TEXTREF("B7", FALSE)` equals the #REF! error value, because "B7" can't be interpreted as an R1C1-style reference.

Related Functions

DEREF Returns the values of the cells in a reference

REFTEXT Converts a reference to text

TEXT.TO.COLUMNS

Equivalent to clicking the Text To Columns command on the Data menu when a column of data is to be separated into multiple columns. Parses text into columns of data.

Syntax

TEXT.TO.COLUMNS(destination_ref, data_type, text_delim, consecutive_delim, tab, semicolon, comma, space, other, other_char, field_info)

The following arguments correspond to check boxes, option buttons and text buttons in the Text To Columns Wizard, which is started with the Text To Columns command on the Data menu.

Destination_Ref is a single cell reference to specifies where to place the converted text.

Data_Type is a number specifying whether that data is delimited (1) or fixed width (2)

Text_Delim denotes how text strings are represented, and can be one of the following values:

Number	Text_Delim
1	"
2	'
3	none

Consecutive_delim is a logical value corresponding to the Treat Consecutive Delimiters as One check box which, if TRUE, allows consecutive delimiters (such as ",,,") to be treated as a single delimiter. If FALSE, all consecutive delimiters are considered separate column breaks.

Tab is a logical value which, if TRUE, specifies that the column has tab delimiters. If FALSE, the column does not have tab delimiters. Tab is ignored if data_type is 2.

Semicolon is a logical value which, if TRUE, specifies that the column has semicolon delimiters. If FALSE, the column does not have semicolon delimiters. Semicolon is ignored if data_type is 2.

Comma is a logical value which, if TRUE, specifies that the column has comma delimiters. If FALSE, the column does not have comma delimiters. Comma is ignored if data_type is 2.

Space is a logical value which, if TRUE, specifies that the column has space delimiters. If FALSE, the column does not have space delimiters. Space is ignored if data_type is 2.

Other is a logical value which, if TRUE, specifies that the column has custom delimiters. If FALSE, the column does not have custom delimiters. Other is ignored if data_type is 2.

Other_Char is a single character that specifies a delimiter not included in the list of delimiters. Other_Char is ignored if data_type is 2 and if the argument other is FALSE.

Field_Info is an array which consists of the following elements: "column number, data_format", if data_type is 1; or "start_pos, data_format" if data_type is 2. The second number defines the column's data format, and can be one of the following.

2nd Number	**Data Format**
1	General
2	Text
3	Date, in the form MDY
4	Date, in the form DMY
5	Date, in the form YMD
6	Date, in the form MYD
7	Date, in the form DYM
8	Date, in the form YDM

9	Do not import column (skip)

TRACER.CLEAR

Equivalent to clicking the Remove All Arrows button on the Auditing toolbar on a worksheet. Clears all tracer arrows on the worksheet.

Syntax

TRACER.CLEAR()

Remark

Returns the #VALUE! error value if not available; for example, the selection is something other than worksheet.

Related Function

TRACER.DISPLAY	Allows tracer arrow to be displayed showing relationship among cells

TRACER.DISPLAY

Equivalent to clicking the Trace Precedents or Trace Dependents buttons on the Auditing toolbar on a worksheet. Allows tracer arrow to be graphically displayed showing relationship among cells.

Syntax

TRACER.DISPLAY(direction, create)

Direction is logical value which, if TRUE, displays tracer arrows for precedents. If FALSE tracer arrows for dependents are displayed.

Create is a logical value which, if TRUE displays the next level of tracer arrows in the direction specified by direction. If FALSE, removes the current level of tracer arrows in the direction specified by direction. A level is the number of "arrows" away from the source cell.

Remark

Returns the #VALUE! error value if not available; for example, the selection is something other than a worksheet, or the cell(s) cannot be traced.

Related Function

TRACER.CLEAR	Clears all tracer arrows on the worksheet

TRACER.ERROR

Equivalent to clicking the Trace Error button on the Auditing toolbar on a worksheet. Allow tracer arrows to be graphically displayed showing which cells are generating an error value.

Syntax

TRACER.ERROR()

Returns TRUE if Microsoft Excel successfully found the cell at which the error occurred. Returns FALSE if an error is not found.

Remark

- Returns the #VALUE! error value if not available; for example, the selection is something other than worksheet, or cell(s) that cannot be traced.
- If you need to know if there is an error in a cell, use ISERROR().

Related Functions

TRACER.DISPLAY	Allows tracer arrow to be displayed
TRACER.CLEAR	Clears all tracer arrows on the worksheet

TRACER.NAVIGATE

Equivalent to double-clicking on a displayed tracer arrow. Moves the selection from one end of a tracer arrow to the other. If it is an error tracer arrow, then the selection goes to the end of the branch.

Syntax

TRACER.NAVIGATE(direction, arrow_num, ref_num)

Direction is a logical value which, if TRUE, moves the selection to the arrow endpoint in the precedents direction. If FALSE, moves the selection to the arrow endpoint in the dependents direction.

Arrow_num is a number specifying which reference a tracer arrow will follow. For example, a 1 indicates that the arrow will follow the first reference in the formula. 1 is the default.

Ref_num If the arrow is an external reference arrow with multiple links, this argument tells which of the links to follow. Refer to the Links dialog, which is displayed with the Links command from the Edit menu. If ref_num is 1, the link in the first reference in the Links dialog box will be followed. The default is 1.

Remarks

- Returns TRUE if successful. Returns FALSE if arrow_num exceeds the number of tracer arrows or if there are no tracer arrows.
- Returns FALSE if ref_num exceeds the number of links.
- Returns the #VALUE! error value if not available; for example, if the selection is something other than a worksheet, or the active cell does not contain an arrow.

Related Function

TRACER.DISPLAY Allows tracer arrow to be displayed showing which cells formulas in other cells depend on

TTESTM

Performs a two-sample Student's t-Test for means, assuming equal variances.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

TTESTM(**inprng1, inprng2**, outrng, labels, alpha, difference)

TTESTM?(inprng1, inprng2, outrng, labels, alpha, difference)

Inprng1 is the input range for the first data set.

Inprng2 is the input range for the second data set.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Labels is a logical value.

- If labels is TRUE, then labels are in the first row or column of the input ranges.
- If labels is FALSE or omitted, all cells in inprng1 and inprng2 are considered data. The output table will include default row or column headings.

Alpha is the confidence level for the test. If omitted, alpha is 0.05.

Difference is the hypothesized difference in means. If omitted, difference is 0.

Related Functions

PTTESTM Performs a paired two-sample Student's t-Test for means

PTTESTV Performs a two-sample Student's t-Test, assuming unequal variances

UNDO

Equivalent to clicking the Undo command on the Edit menu. Reverses certain actions and commands. UNDO is available in the same situations as the Undo command.

Syntax

UNDO( )

UNGROUP

Separates a grouped object into individual objects. Use UNGROUP to separate a group of objects so that you can format, move, or size one of the objects.

If the selection is not a grouped object, UNGROUP returns FALSE.

Syntax

UNGROUP( )

Related Function

GROUP   Groups selected objects

UNHIDE

Equivalent to clicking the Unhide command on the Window menu. Use UNHIDE to display hidden windows.

Syntax

UNHIDE(**window_text**)

Window_text   is the name of the window to unhide. If window_text is not the name of an open workbook, an error value is returned and the macro is interrupted. You cannot unhide a window of an add-in workbook.

Tip   You can use UNHIDE to activate an embedded chart in order to edit and format it. Use the HIDE function to de-activate the chart window.

Related Functions

GET.WINDOW   Returns information about a window

HIDE   Hides the active window

UNLOCKED.NEXT, UNLOCKED.PREV

Equivalent to pressing TAB or SHIFT+TAB to move to the next or previous unlocked cell in a protected worksheet. Use these functions when you want to control which cell is active on a protected sheet.

Syntax

UNLOCKED.NEXT( )

UNLOCKED.PREV( )

Related Functions

CELL.PROTECTION   Controls protection for the selected cells

PROTECT.DOCUMENT   Controls protection for the active sheet

UNREGISTER

Unregisters a previously registered dynamic link library (DLL) or code resource. You can use UNREGISTER to free memory that was allocated to a DLL or code resource when it was registered. There are two syntax forms of this function. Use syntax 1 when you want Microsoft Excel to unregister a function or code resource according to its use count. Use syntax 2 when you want Microsoft Excel to unregister a function or code resource regardless of the use count.

Syntax 1

UNREGISTER(**register_id**)

Register_id   is the register ID returned by the REGISTER or REGISTER.ID function, which corresponds to the function or code resource to be removed from memory.

Microsoft Excel counts the number of times you register a function or code resource. This number is called the use count. Each time you unregister a function or code resource, its use count is decremented by 1. When the use count equals 0, Microsoft Excel frees the allocated memory. Therefore, if you register a function or code resource more than once, you must use a corresponding number of UNREGISTER functions to ensure that it is completely unregistered.

Note Because Microsoft Excel for Windows and Microsoft Excel for the Macintosh use different types of code resources, UNREGISTER has a slightly different syntax form when used in each operating environment.

Syntax 2a

For Microsoft Excel for Windows

UNREGISTER(**module_text**)

Syntax 2b

For Microsoft Excel for the Macintosh

UNREGISTER(**file_text**)

Module_text or file_text is text specifying the name of the dynamic link library (DLL) that contains the function (in Microsoft Excel for Windows) or the name of the file that contains the code resource (in Microsoft Excel for the Macintosh).

If you use this syntax of UNREGISTER, all functions in the DLL (or all code resources in the file) are immediately unregistered, regardless of the use count.

Examples

Assuming that a REGISTER function in cell A5 of a macro sheet has already run (and has run only once), the following macro formula unregisters the corresponding function or code resource:

```
UNREGISTER(A5)
```

You could also use REGISTER.ID to return the register ID, instead of specifying a cell reference:

```
UNREGISTER(REGISTER.ID("User", "GetTickCount")
```

Assuming that you have registered several different functions from the USER.EXE DLL of Microsoft Windows, the following macro formula unregisters all functions in that DLL:

```
UNREGISTER("User")
```

Tip If you register a function or code resource, and use the optional function_text argument to specify a custom name that will appear in the Paste Function dialog box, this custom name will not be removed by the UNREGISTER function. To remove the custom name, use the SET.NAME function without its second argument.

Related Function

REGISTER Registers a code resource

UPDATE.LINK

Equivalent to clicking the Links command on the Edit menu and clicking the Update Now button with a link selected in the Links dialog box. Updates a link to another document. Use UPDATE.LINK to get the newest information from a supporting document.

Syntax

UPDATE.LINK(link_text, type_of_link)

Link_text is a text string describing the full path of the link as displayed in the Links dialog box. If link_text is omitted, only links from the active workbook to other Microsoft Excel workbooks are updated.

Type_of_link is a number from 1 to 4 that specifies the type of link to update.

Type_of_link	**Link document type**
1 or omitted	Microsoft Excel link
2	DDE link
3	Not available
4	Not available

Related Functions

CHANGE.LINK	Changes supporting links
GET.LINK.INFO	Returns information about a link
OPEN.LINKS	Opens specified supporting documents

VBA.INSERT.FILE

Inserts a text file containing code directly into your Visual Basic module.

Syntax

VBA.INSERT.FILE(**filename_text**)

Filename_text is the name of a text file that contains Microsoft Visual Basic code that is inserted into the currently active module.

VBA.MAKE.ADDIN

Converts a workbook containing Visual Basic procedures into an add-in.

Syntax

VBA.MAKE.ADDIN(**filename_text**)

Filename_text   is the name of the workbook that you want to convert to an add-in.

Remarks

For information about creating add-ins with Visual Basic, see Chapter 13, "Creating Automatic Procedures and Add-in Applications" in the Visual Basic User's Guide.

VIEW.3D

Equivalent to clicking the 3-D View command on the Format menu in Microsoft Excel version 4.0, available when a chart sheet is the active sheet. Adjusts the view of the active 3-D chart. Use VIEW.3D to emphasize different parts of your chart by viewing it from different angles and perspectives.

Syntax

VIEW.3D(elevation, perspective, rotation, axes, height%, autoscale)

VIEW.3D?(elevation, perspective, rotation, axes, height%, autoscale)

Elevation   is a number from -90 to 90 specifying the viewing elevation of the chart and is measured in degrees. Elevation corresponds to the Elevation box in the 3-D View dialog box in Microsoft Excel version 4.0.

- If elevation is 0, you view the chart straight on. If elevation is 90, you view the chart from above (a "bird's eye view"). If elevation is -90, you view the chart from below.
- If elevation is omitted, the current value is used..
- Elevation is limited to 0 to 44 for 3-D bar charts and 0 to 80 for 3-D pie charts.

Perspective   is a number from 0 to 100% specifying the perspective of the chart. Perspective corresponds to the Perspective box in the 3-D View dialog box in Microsoft Excel version 4.0.

- A higher perspective value simulates a closer view.
- If perspective is omitted, the current value is used..
- Perspective is ignored on 3-D bar and pie charts.

Rotation   is a number from 0 to 360 specifying the rotation of the chart around the value (z) axis and is measured in degrees. Rotation corresponds to the Rotation box in the 3-D View dialog box in Microsoft Excel version 4.0. As you rotate the chart, the back and side walls are moved so that they do not block the chart.

- If rotation is omitted, the current value is used..
- Rotation is limited to 0 to 44 for 3-D bar charts.

Axes   is a logical value specifying whether axes are fixed in the plane of the screen or can rotate with the chart. Axes corresponds to the Right Angle Axes check box in the 3-D View dialog box in Microsoft Excel version 4.0.

- If axes is TRUE, Microsoft Excel locks the axes.
- If axes is FALSE, Microsoft Excel allows the axes to rotate.
- If axes is omitted and the chart view is 3-D layout, axes is assumed to be FALSE.
- Axes is TRUE for 3-D bar charts and ignored for 3-D pie charts.

Height%   is a number from 5 to 500 specifying the height of the chart as a percentage of the length of the base. Height% corresponds to the Height box in the 3-D View dialog box in Microsoft Excel version 4.0. Height% is useful for controlling the appearance of charts with many series or data points. If height% is omitted, the current value is used..

Autoscale   is a logical value corresponding to the Auto Scaling check box in the 3-D View dialog box in Microsoft Excel version 4.0. If TRUE, automatic scaling is used; if FALSE, it is not; if omitted, the current setting is not changed.

Related Function

FORMAT.MAIN   Formats a main chart

VIEW.DEFINE

Equivalent to clicking the Add button in the Custom Views dialog box in Microsoft Excel 97 or later, which appears when you click the Custom Views command on the View menu. In Microsoft Excel 97, the Custom Views command replaced the View Manager command that was available in Microsoft Excel 95 and earlier versions. Creates or replaces a view.

If this function is not available in Microsoft Excel 95 or in earlier versions, you must install the View Manager add-in.

Syntax

VIEW.DEFINE(**view_name**, print_settings_log, row_col_log)

View_name   is text enclosed in quotation marks and specifies the name of the view you want to define for the sheet. If the workbook already contains a view with view_name, the new view replaces the existing one.

Print_settings_log   is a logical value that, if TRUE or omitted, includes current print settings in the view or, if FALSE, does not include current print settings in the view.

Row_col_log   is a logical value that, if TRUE or omitted, includes current row and column settings in the view or, if FALSE, does not include current row and column settings in the view.

Related Functions

VIEW.DELETE Removes a view from the active workbook

VIEW.SHOW Shows a view

VIEW.DELETE

Equivalent to selecting a view and clicking the Delete button in the Custom Views dialog box, which appears when you click the Custom Views command on the View menu. In Microsoft Excel 97 or later, the Custom Views command replaced the View Manager command that was available in Microsoft Excel 95 and earlier versions. Removes a view from the active workbook.

If this function is not available in Microsoft Excel 95 or in earlier versions, you must install the View Manager add-in.

Syntax

VIEW.DELETE(**view_name**)

View_name is text enclosed in quotation marks and specifies the name of the view in the current workbook that you want to delete.

Remarks

VIEW.DELETE returns the #VALUE error value if view_name is invalid or if the workbook is protected.

Related Functions

VIEW.DEFINE Creates or replaces a view

VIEW.SHOW Shows a view

VIEW.GET

Equivalent to displaying a list of views in the Custom Views dialog box, which appears when you click the Custom Views command on the View menu. In Microsoft Excel 97 or later, the Custom Views command replaces the View Manager command that was available in Microsoft Excel 95 and earlier versions. Returns an array of views from the active workbook.

If this function is not available in Microsoft Excel 95 or in earlier versions, you must install the View Manager add-in.

Syntax

VIEW.GET(**type_num**, view_name)

Type_num is a number from 1 to 3 that specifies the type of information to return, as shown in the following table.

Type_num	**Result**
1	Returns an array of views from all the sheets in the active workbook or the #N/A error value if none are defined.

2	Returns TRUE if print settings are included in the specified view. Returns FALSE if print settings are not included. Returns the #VALUE! error value if the name is invalid or the workbook is protected.
3	Returns TRUE if row and column settings are included in the specified view. Returns FALSE if row and column settings are not included. Returns the #VALUE! error value if the name is invalid or the workbook is protected.

View_name is text enclosed in quotation marks and specifies the name of a view in the active workbook. View_name is required if type_num is 2 or 3.

Example

The following macro formula returns an array of views from the active workbook:

```
VIEW.GET(1)
```

Related Functions

VIEW.DEFINE Creates or replaces a view

VIEW.DELETE Removes a view from the active workbook

VIEW.SHOW Shows a view

VIEW.SHOW

Equivalent to selecting a view and clicking the Show button in the Custom Views dialog box, which appears when you click the Custom Views command on the View menu. In Microsoft Excel 97 or later, the Custom Views command replaces the View Manager command that was available in Microsoft Excel 95 and earlier versions. Shows a view.

If this function is not available in Microsoft Excel 95 or in earlier versions, you must install the View Manager add-in.

Syntax

VIEW.SHOW(**view_name**)

VIEW.SHOW?(view_name)

View_name is text enclosed in quotation marks and specifies the name of a view in the active workbook.

Remarks

VIEW.SHOW returns the #VALUE error value if view_name is invalid or the workbook is protected.

Related Functions

VIEW.DEFINE Creates or replaces a view

VIEW.DELETE Removes a view from the active workbook

VLINE

Scrolls through the active window vertically by the number of rows you specify.

Syntax

VLINE(**num_rows**)

Num_rows   is a number that specifies how many rows to scroll.

- If num_rows is positive, Microsoft Excel scrolls down by the number of rows indicated by num_rows.
- If num_rows is negative, Microsoft Excel scrolls up by the number of rows indicated by num_rows.

Related Functions

HLINE   Horizontally scrolls through the active window by columns

HPAGE   Horizontally scrolls through the active window one window at a time

HSCROLL   Horizontally scrolls through a worksheet by percentage or by column number

VPAGE   Vertically scrolls through the active window one window at a time

VSCROLL   Vertically scrolls through a worksheet by percentage or by row number

VOLATILE

Specifies whether a custom worksheet function is volatile or nonvolatile. A volatile custom function is recalculated every time a calculation occurs on the worksheet.

Syntax

VOLATILE(logical)

Logical   is a logical value specifying whether the custom function is volatile or nonvolatile. If logical is TRUE or omitted, the function is volatile; if FALSE, nonvolatile.

Remarks

- VOLATILE must precede every other formula in the custom function except RESULT and ARGUMENT.
- Normally, a worksheet recalculates a cell containing a nonvolatile custom function only when any part of the complete formula in the cell is recalculated. Use VOLATILE(TRUE) to recalculate the function every time the worksheet is recalculated.
- Most custom functions are nonvolatile by default, but custom functions with reference arguments are volatile by default. Use VOLATILE(FALSE) to prevent these functions from being recalculated unnecessarily often.

Related Function

RESULT   Specifies the data type a custom function returns

VPAGE

Vertically scrolls through the active window one window at a time. Use VPAGE to change the displayed area of a worksheet or macro sheet.

Syntax

VPAGE(**num_windows**)

Num_windows   specifies the number of windows to scroll through the active window vertically. A window is defined as the number of visible rows. If 20 rows are visible in the window, VPAGE scrolls in increments of 20 rows.

- If num_windows is positive, VPAGE scrolls down.
- If num_windows is negative, VPAGE scrolls up.

Related Functions

HPAGE   Horizontally scrolls through the active window one window at a time

HSCROLL   Horizontally scrolls through a worksheet by percentage or by column number

VLINE   Vertically scrolls through the active window by rows

VSCROLL   Vertically scrolls through a worksheet by percentage or by row number

VSCROLL

Vertically scrolls through the active sheet by percentage or by row number.

Syntax

VSCROLL(**position**, row_logical)

Position   specifies the row you want to scroll to. Position can be an integer representing the row number or a fraction or percentage representing the vertical position of the row in the sheet. If position is 0, VSCROLL scrolls through your sheet to its top edge, which is row 1. If position is 1, VSCROLL scrolls through your sheet to its bottom edge, which is row 16, 384 in Microsoft Excel 95 or earlier, or row 65,536 in Microsoft Excel 97 or later. For charts that do not size with the window, use a fraction or percentage.

Row_logical   is a logical value specifying how the function scrolls.

- If row_logical is TRUE, VSCROLL scrolls through the sheet to row position.
- If row_logical is FALSE or omitted, VSCROLL scrolls through the sheet to the vertical position represented by the fraction position.

Remarks

- To scroll to a specific row n, either use VSCROLL(n, TRUE) or VSCROLL(n/16384) in Microsoft Excel 95 or earlier; in Microsoft Excel 97 or later, you should use VSCROLL(n/65536). To scroll to row 138, for example, enter VSCROLL(138, TRUE) (in any version) or VSCROLL(138/16384) in earlier versions of Microsoft Excel or VSCROLL(138/65536) in Microsoft Excel 97 or later

- If you are recording a macro and move the scroll box several times in a row, the recorder only records the final location of the scroll box, omitting any intermediate steps. Remember that scrolling does not change the active cell or the selection.

Related Functions

FORMULA.GOTO   Selects a named area or reference on any open workbook

HLINE   Horizontally scrolls through the active window by columns

HPAGE   Horizontally scrolls through the active window one window at a time

HSCROLL   Horizontally scrolls through a sheet by percentage or by column number

SELECT   Selects a cell, object, or chart item

VLINE   Vertically scrolls through the active window by rows

VPAGE   Vertically scrolls through the active window one window at a time

WAIT

Pauses the macro until the time specified by the serial number.

Syntax

WAIT(**serial_number**)

Serial_number   is the date-time code used by Microsoft Excel for date and time calculations. You can give serial_number as text, such as "4:30 PM", or as a formula, such as NOW()+"00:00:04", instead of as a number. The text or formula is automatically converted to a serial number. For more information about serial_number, see NOW.

Important   WAIT suspends all Microsoft Excel activity and may prevent you from performing other operations on your computer. Background processes, such as printing and recalculation, are continued.

Example

Use WAIT with NOW to pause a macro for a length of time or until the time specified by the serial number. For example, the following macro formula waits 3 seconds from the time the functions are evaluated:

```
WAIT(NOW()+"00:00:03")
```

Related Function

ON.TIME   Runs a macro at a specific time

WHILE

Carries out the statements between the WHILE function and the next NEXT function until logical_test is FALSE. Use WHILE-NEXT loops to carry out a series of macro formulas while a certain condition remains TRUE.

Syntax

WHILE(**logical_test**)

Logical_test is a value or formula that evaluates to TRUE or FALSE. If logical_test is FALSE the first time the WHILE function is reached, the macro skips the loop and resumes running at the statement after the next NEXT function. If there is no NEXT function in the same column, WHILE displays an error message and interrupts the macro.

Remarks

If you know exactly how many times you'll need to carry out the statements within a loop, in most cases you should use a FOR-NEXT loop. Also, avoid creating an infinite loop by making sure logical_test does not always evaluate to TRUE.

Example

The following statement starts a loop that executes while the value in the current cell is less than 5:

```
=WHILE(TYPE(ACTIVE.CELL()<5))
```

The following statement starts a loop that executes until the position in the open file identified as FileNumber reaches the end of the file:

```
=WHILE(FPOS(FileNumber)<=FSIZE(FileNumber))
```

Related Functions

FOR Starts a FOR-NEXT loop

FOR.CELL Starts a FOR.CELL-NEXT loop

IF Specifies an action to take if a logical test is TRUE

NEXT Ends a FOR-NEXT, FOR.CELL-NEXT, or WHILE-NEXT loop

WINDOW.MAXIMIZE

Changes the active window from its normal size to full size. In Microsoft Excel for Windows, using WINDOW.MAXIMIZE is equivalent to pressing CTRL+F10 or double-clicking the title bar. In Microsoft Excel for the Macintosh, using WINDOW.MAXIMIZE is equivalent to double-clicking the title bar or clicking the zoom box.

Syntax

WINDOW.MAXIMIZE(window_text)

Window_text specifies which window to switch to and maximize. Window_text is text enclosed in quotation marks or a reference to a cell containing text. If window_text is omitted, the active window is maximized.

Remarks

WINDOW.MAXIMIZE replaces FULL(TRUE) in earlier versions of Microsoft Excel.

Related Functions

WINDOW.MINIMIZE	Minimizes a window
WINDOW.MOVE	Moves a window
WINDOW.RESTORE	Restores a window to its previous size
WINDOW.SIZE	Changes the size of a window

WINDOW.MINIMIZE

Shrinks a window to an icon. In Microsoft Excel for Windows, using WINDOW.MINIMIZE is equivalent to clicking the minimize button on a workbook window. In Microsoft Excel for the Macintosh, the minimize feature is not supported.

Syntax

WINDOW.MINIMIZE(window_text)

Window_text specifies which window to minimize.

- Window_text is text enclosed in quotation marks or a reference to a cell containing text.
- If window_text is omitted, Microsoft Excel minimizes the active window.

Remarks

If a window is already minimized, WINDOW.MINIMIZE has no effect.

Related Functions

WINDOW.MAXIMIZE	Maximizes a window
WINDOW.MOVE	Moves a window
WINDOW.RESTORE	Restores a window to its previous size
WINDOW.SIZE	Changes the size of a window

WINDOW.MOVE

Equivalent to clicking the Move command on the Control menu in Microsoft Excel for Windows or moving a window by dragging its title bar or its icon. Moves the active window so that its upper-left corner is at the specified horizontal and vertical positions. The dialog-box form, WINDOW.MOVE?, is supported only in Microsoft Excel for Windows.

Syntax

WINDOW.MOVE(x_pos, y_pos, window_text)

WINDOW.MOVE?(x_pos, y_pos, window_text)

X_pos is the horizontal position to which you want to move the window. X_pos is measured in points. A point is 1/72nd of an inch.

- In Microsoft Excel for Windows, x_pos is measured from the left edge of your workspace to the left edge of the window.
- In Microsoft Excel for the Macintosh, x_pos is measured from the left edge of your screen to the left edge of the window.
- If x_pos is omitted, the window does not move horizontally.

Y_pos is the vertical position to which you want to move the window. Y_pos in measured in points from the bottom edge of the formula bar to the top edge of the window. If y_pos is omitted, the window does not move vertically.

Window_text specifies which window to restore.

- Window_text is text enclosed in quotation marks or a reference to a cell containing text.
- If window_text is omitted, it is assumed to be the name of the active window.

Remarks

- If the window is minimized, WINDOW.MOVE moves the icon on the workspace. Measurements are relative to the upper-left corner of the workspace and the icon.
- WINDOW.MOVE does not change the size of the window or affect whether the specified window is active or inactive.
- In Microsoft Excel for the Macintosh, if window_text is "Clipboard", WINDOW.MOVE moves the Clipboard. The Clipboard must already be available; if it is not available, use the SHOW.CLIPBOARD function before using the WINDOW.MOVE function.
- WINDOW.MOVE replaces MOVE in earlier versions of Microsoft Excel.

Related Functions

FORMAT.MOVE Moves the selected object

WINDOW.MAXIMIZE Maximizes a window

WINDOW.MINIMIZE Minimizes a window

WINDOW.RESTORE Restores a window to its previous size

WINDOW.SIZE Changes the size of a window

WINDOW.RESTORE

Changes the active window from maximized or minimized size to its previous size. In Microsoft Excel for Windows, using WINDOW.RESTORE is equivalent to pressing CTRL+F5 or double-clicking the title bar (or double-clicking the icon if it is minimized). In Microsoft Excel for the Macintosh, using WINDOW.RESTORE is equivalent to double-clicking the title bar or clicking the zoom box.

Syntax

WINDOW.RESTORE(window_text)

Window_text specifies which window to switch to and restore.

- Window_text is text enclosed in quotation marks or a reference to a cell containing text.
- If window_text is omitted, Microsoft Excel restores the active window.

Remarks

- If the window is minimized, WINDOW.RESTORE restores the icon to its previous size. This operation is equivalent to double-clicking the icon.
- WINDOW.RESTORE replaces FULL(FALSE) in earlier versions of Microsoft Excel.

Related Functions

WINDOW.MAXIMIZE Maximizes a window

WINDOW.MINIMIZE Minimizes a window

WINDOW.MOVE Moves a window

WINDOW.SIZE Changes the size of a window

WINDOWS

Returns the names of the specified open Microsoft Excel windows, including hidden windows. Use WINDOWS to get a list of active windows for use by other macro functions that return information about or manipulate windows, such as GET.WINDOW and ACTIVATE. The names are returned as a horizontal array of text values, in order of their appearance on your screen. The first name is the active window, the next name is the window directly under the active window, and so on.

Syntax

WINDOWS(type_num, match_text)

Type_num is a number that specifies which types of workbooks are returned by WINDOWS, according to the following table.

Type_num	Returns window names from these types of documents
1 or omitted	All windows except those belonging to add-in workbooks
2	Add-in workbooks only
3	All types of workbooks

Match_text specifies the windows whose names you want returned and can include wildcard characters. If match_text is omitted, WINDOWS returns the names of all open windows.

Tips

- You can change the output of a horizontal array to vertical with the TRANSPOSE function.
- You can use WINDOWS with the INDEX function to select individual window names from the array for use in other functions that take window names as arguments.
- You can use the COLUMNS functions to count the number of entries in the array, which is the number of windows.

Examples

If the active window, named BOOK1, is on top of a window named MACROS:2, which is on top of a window named MACROS:1, then:

`WINDOWS()` equals {"BOOK1", "MACROS:2", "MACROS:1"}

Related Functions

ACTIVATE	Switches to a window
DOCUMENTS	Returns the names of the specified open workbooks
GET.WINDOW	Returns information about a window
NEW.WINDOW	Creates a new window for an existing sheet or macro sheet
ON.WINDOW	Runs a macro when you switch to a window

WINDOW.SIZE

Equivalent to clicking the Size command on the Control menu or to adjusting the sizing borders (in Microsoft Excel for Windows) or the sizing box (in Microsoft Excel for the Macintosh) of the window with the mouse. Changes the size of the active window by moving its lower-right corner so that the window has the width and height you specify. WINDOW.SIZE does not change the position of the upper-left corner of the window, nor does it affect whether the specified window is active or inactive.

Syntax

WINDOW.SIZE(**width, height**, window_text)

WINDOW.SIZE?(width, height, window_text)

Width specifies the width of the window and is measured in points. A point is 1/72nd of an inch.

Height specifies the height of the window and is measured in points.

Window_text specifies which window to size.

- Window_text is text enclosed in quotation marks or a reference to a cell containing text.
- If window_text is omitted, it is assumed to be the name of the active window.

Remarks

- In Microsoft Excel for Windows, an error occurs if you try to resize a window that has already been minimized to an icon or enlarged to its maximum size. You must first restore the window to its original size using the WINDOW.RESTORE function. For more information, see WINDOW.RESTORE.
- WINDOW.SIZE replaces SIZE in earlier versions of Microsoft Excel.

Related Functions

FORMAT.SIZE	Sizes an object
WINDOW.MAXIMIZE	Maximizes a window
WINDOW.MINIMIZE	Minimizes a window
WINDOW.MOVE	Moves a window
WINDOW.RESTORE	Restores a window to its previous size

WINDOW.TITLE

Changes the title of the active window to the title you specify. The title appears at the top of the workbook window. Use WINDOW.TITLE to control window titles when you're using Microsoft Excel to create a custom application.

Syntax

WINDOW.TITLE(text)

Text   is the title you want to assign to the window. If text is omitted, it is assumed to be the name of the workbook as it is stored on your disk. Empty text ("") specifies no title.

Important   WINDOW.TITLE changes the name of the window, not the actual name of the workbook as it is stored on your disk. To change the name of the workbook, use the SAVE.AS function.

Remarks

- The window name you create using WINDOW.TITLE will appear on the Window menu, and will be returned by the WINDOWS function, but not by the DOCUMENTS function. You must use the new window name in theACTIVATE function and the ON.WINDOW function.
- If you want to communicate with a Microsoft Excel workbook using DDE functions like INITIATE or REQUEST, you must specify the filename of the workbook and not the window title specified with the WINDOW.TITLE function.
- If you use NEW.WINDOW to create new windows on the workbook, the window title will be restored to its original name.

Example

The following macro formula changes the title of the active window to First Quarter.

```
WINDOW.TITLE("First Quarter")
```

Related Functions

APP.TITLE   Changes the title of the application workspace

SAVE.AS   Specifies a new filename, file type, protection password, or write-reservation password, or to create a backup file

WORKBOOK.ACTIVATE

Equivalent to activating a worksheet by clicking on its tab.

Syntax

WORKBOOK.ACTIVATE(**sheet_name**)

Sheet_name   is the name of the sheet you want to activate. You can use WORKBOOK.ACTIVATE to activate a sheet within the active workbook, or you can also activate a sheet in another workbook by using the [workbook]sheet_name reference for the sheet_name argument.

Related Functions

WORKBOOK.SELECT   Selects one or more sheets for group editing

WORKBOOK.OPTIONS   Changes the settings of a workbook sheet

WORKBOOK.ADD

Macro SheetsOnly

Equivalent to clicking the Add button in the workbook contents window in Microsoft Excel version 4.0. Moves a sheet between workbooks. For Microsoft Excel version 5.0 or later use WORKBOOK.MOVE.

Syntax

WORKBOOK.ADD(**name_array**, dest_book, position_num)

WORKBOOK.ADD?(name_array, dest_book, position_num)

Name_array   is the name of a sheet, or an array of names of sheets, that you want to move.

Dest_book   is the name of the workbook to which you want to add name_array. If dest_book is omitted, it is assumed to be the active workbook.

Position_num   is a number that specifies the position of the sheet within the workbook.

Related Functions

WORKBOOK.MOVE   Moves one or more sheets between workbooks or changes a sheet's position within a workbook

WORKBOOK.COPY   Copies one or more documents from their current workbook to another workbook

WORKBOOK.COPY

Equivalent to clicking the Move or Copy Sheet command on the Edit menu. Copies one or more sheets from their current positions to the specified position in the same workbook or to another workbook.

Syntax

WORKBOOK.COPY(**name_array**, dest_book, position_num)

WORKBOOK.COPY?(name_array, dest_book, position_num)

Name_array is the name of a sheet, or an array of names of sheets, that you want to copy in the currently active workbook.

Dest_book is the name of the workbook to which you want to copy name_array. If dest_book is the current workbook, name_array is copied within the workbook. If dest_book is omitted, the copy of name_array becomes a separate workbook.

Position_num is a number that specifies the target position for the sheet within the new workbook. The first position is 1.

- If position_num is specified, Microsoft Excel inserts the copy of the sheet at the specified position in the workbook.
- If position_num is omitted, Microsoft Excel places the sheet at the last position in the workbook.
- If dest_book is omitted, position_num is ignored.

Remarks

- If the structure of the workbook is protected, you cannot copy sheets within the workbook or to another workbook.
- You cannot copy a hidden sheet.

Related Function

WORKBOOK.MOVE Moves one or more documents from one workbook to another workbook or to another position in the same workbook

WORKBOOK.DELETE

Equivalent to clicking the Delete Sheet command on the Edit menu. Deletes a sheet or group of sheets from the current workbook.

Syntax

WORKBOOK.DELETE(sheet_text)

Sheet_text is the name of the sheet to delete. If omitted, the currently active sheet or sheets is deleted.

Remarks

- This function prompts for confirmation. To suppress the prompt, use the ERROR function. For example, =ERROR(FALSE).

- If the structure of the workbook is protected, you cannot delete any of its sheets.
- If you want to delete Sheet1:Sheet10, you must select them first with WORKBOOK.SELECT(). You can also place the sheets in an array first, as in {"Sheet1", "Sheet2", "Sheet3",...}.
- You cannot delete the last visible sheet in a workbook.

WORKBOOK.HIDE

Equivalent to clicking the Sheet command on the Format menu, and then clicking Hide on the Sheet submenu. Hides sheets in the active workbook.

Syntax

WORKBOOK.HIDE(sheet_text, very_hidden)

Sheet_text is the name of the sheet to hide. If omitted, the currently selected sheet(s) are hidden.

Very_hidden specifies how the sheet is hidden. If TRUE, then the sheet name does not appear in the Unhide dialog box. After using this argument, use WORKBOOK.UNHIDE to unhide the sheet. If FALSE or omitted, hides the sheet but does not prevent the sheet's name from appearing in the Unhide dialog box.

Remarks

- If the structure of the workbook is protected, you cannot hide any sheets in the workbook.
- You cannot hide the last visible sheet in a workbook.
- To hide Sheet1:Sheet10, select them first with the WORKBOOK.SELECT function. You can also place the sheets in an array first, as in {"Sheet1", "Sheet2", "Sheet3",...}.

WORKBOOK.INSERT

Equivalent to clicking the Worksheet, Chart, or Macro commands on the Insert menu. Inserts one or more new sheets into the current workbook.

Syntax

WORKBOOK.INSERT(**type_num**)

WORKBOOK.INSERT?(**type_num**)

Type_num specifies the type of sheet to insert.

Type_num	Type of sheet
1	Worksheet

2	Chart
3	Microsoft Excel 4.0 Macro Sheet
4	Microsoft Excel 4.0 International Macro Sheet
5	(Reserved)
6	Microsoft Excel Visual Basic Module
7	Dialog
Quoted text	Template

If omitted, the type of the active sheet is used.

If the current selection contains one sheet, then only one sheet is inserted. If the selection contains more than one sheet and the active sheet is a worksheet, then an equal number of sheets is inserted to the left of the selected group of sheets.

Remarks

- The new sheets are always inserted to the left of the current selection.
- If the workbook structure is protected, you cannot insert new sheets.

WORKBOOK.MOVE

Equivalent to clicking the Move or Copy Sheet command on the Edit menu. Moves one or more sheets between workbooks or changes a sheet's position within a workbook.

Syntax

WORKBOOK.MOVE(**name_array**, dest_book, position_num)

WORKBOOK.MOVE?(name_array, dest_book, position_num)

Name_array   is the name of a sheet or an array of names of sheets in the active workbook that you want to move.

Dest_book   is the name of the workbook to which you want to move name_array. If dest_book is omitted, WORKBOOK.MOVE moves the sheet out of the workbook and puts it in a new separate workbook. If dest_book is the same as the current book, then the sheet is moved within the workbook.

Position_num   is a number that specifies the target position for the sheet within dest_book. The first position is 1.

- If position_num is specified, Microsoft Excel inserts the sheet at the specified position in the workbook.
- If position_num is omitted, Microsoft Excel moves the sheet to the last position in the workbook. If you move the last sheet out of a workbook, the workbook closes.

Related Function

WORKBOOK.COPY   Copies one or more documents from their current workbook into another workbook

WORKBOOK.NAME

Equivalent to clicking the Rename command on the Sheet submenu of the Format menu. Renames a sheet in a workbook.

Syntax

WORKBOOK.NAME(**oldname_text**, **newname_text**)

WORKBOOK.NAME?(oldname_text, newname_text)

Oldname_text   is the name of the sheet that you want to rename.

Newname_text   is the new name of the sheet.

Remarks

- If you try to rename a sheet using a sheet name that already exists in the workbook, Microsoft Excel displays an error message and interrupts the macro.
- If the structure of the workbook is protected, you cannot rename any of the sheets in the workbook.

WORKBOOK.NEW

Adds a sheet to a workbook. This function is for compatibility with Microsoft Excel version 4.0. To add a new sheet to a workbook in Microsoft Excel version 5, use the WORKBOOK.INSERT function.

Syntax

WORKBOOK.NEW( )

WORKBOOK.NEW?( )

In both forms of this function, you will be prompted for the name of the workbooks.

Related Function

WORKBOOK.INSERT   Adds sheets to workbooks

WORKBOOK.NEXT

Activates the next sheet in the active workbook.

Syntax

WORKBOOK.NEXT( )

Remarks

- If the last sheet in the workbook is active, this function has no effect.
- This function skips over hidden sheets in the workbook.

WORKBOOK.OPTIONS

Equivalent to selecting the Options button in a workbook contents window in Microsoft Excel version 4.0. This function is for compatibility with Microsoft Excel version 4.0. To change the name of a sheet in Microsoft Excel version 5.0, use the WORKBOOK.NAME function.

Syntax

WORKBOOK.OPTIONS(**sheet_name**, bound_logical, new_name)

Sheet_name   is the name of a sheet.

Bound_logical   is for compatibility with Microsoft Excel version 4.0. In Microsoft Excel version 5.0 and later versions, this should be TRUE or omitted because FALSE returns an error.

New_name   is the sheet name to assign to sheet_name. If new_name is omitted, then the sheet name is not changed.

Related Functions

GET.WORKBOOK	Returns information about a workbook sheet
WORKBOOK.NAME	Changes the name of a sheet in a workbook
WORKBOOK.SELECT	Selects the specified sheets in a workbook

WORKBOOK.PREV

Activates the previous sheet in the workbook.

Syntax

WORKBOOK.PREV( )

Remarks

- If the first sheet in the workbook is active, this function has no effect.
- This function skips over hidden sheets in the workbook.

WORKBOOK.PROTECT

Equivalent to clicking the Protect Workbook command on the Protection submenu of the Tools menu. Controls protection of workbooks.

Syntax

WORKBOOK.PROTECT(structure, windows, password)

WORKBOOK.PROTECT?(structure, windows, password)

Structure specifies whether the structure of the workbook is protected. If TRUE, the structure is protected. If FALSE or omitted, the structure is not protected.

Windows specifies whether the windows of the workbook are protected. If TRUE, the windows are protected. If FALSE or omitted, the windows are not protected.

Password specifies whether to protect the workbook with a password. If omitted no password is used. When recording a macro, this argument is not recorded. In the dialog box form of this function, you can specify a password.

Remarks

To protect a sheet in a workbook, use the PROTECT.DOCUMENT function.

Related Function

PROTECT.DOCUMENT	Protects a sheet in a workbook

WORKBOOK.SCROLL

Scrolls through the sheets in a workbook.

Syntax

WORKBOOK.SCROLL(**num_sheets**, firstlast_logical)

Num_sheets. is a number specifying how many sheets to scroll and the direction of scrolling. Positive numbers scroll forward by that many sheets. Negative numbers scroll backward by that many sheets. Zero (0) does not scroll.

Firstlast_logical specifies whether to scroll to the first or last sheet in the workbook. If TRUE, scrolls to the last sheet in the workbook. If FALSE, scrolls to the first sheet in the workbook. If this argument is specified, then num_sheets is ignored.

WORKBOOK.SELECT

Equivalent to selecting a sheet or group of sheets in the active workbook. If you select a group of sheets, subsequent commands effect all the sheets in the group.

Syntax

WORKBOOK.SELECT(name_array, active_name, replace)

Name_array is a horizontal array of text names of sheets you want to select. If name_array is omitted, no sheets are selected.

Active_name is the name of a single sheet in the workbook that you want to be the active sheet. If active_name is omitted, the first sheet in name_array is made the active sheet.

Replace specifies whether the currently selected sheets or macro sheets are to be replaced by name_array. If TRUE or omitted, then the current sheet selection is replaced by name_array. If FALSE, then name_array will be appended to the current sheet.

Related Functions

GET.WORKBOOK Returns information about a workbook

SELECT Selects a cell, worksheet object, or chart item

WORKBOOK.TAB.SPLIT

Sets the ratio of the tabs to the horizontal scrollbar.

Syntax

WORKBOOK.TAB.SPLIT(ratio_num)

Ratio_num is the ratio of tabs to horizontal scroll, as a decimal value between 0 and 1. If omitted defaults to 6.

Remarks

- If the structure of the workbook is protected, you cannot use this function.
- Use GET.WINDOW(28) to find out what the current ratio is.

Related Function

GET.WINDOW Returns information about a workbook window

WORKBOOK.UNHIDE

Equivalent to clicking the Unhide command on the sheet submenu of the Format menu. Unhides one or more sheets in the current workbook.

Syntax

WORKBOOK.UNHIDE(sheet_text)

WORKBOOK.UNHIDE?(sheet_text)

Sheet_text. specifies the sheet that you want to unhide. If sheet_text is omitted, then this function unhides the sheets in the order that they would appear in the workbook.

Remarks

If the workbook is protected, you cannot unhide any sheets in the book.

Related Function

WORKBOOK.HIDE Hides sheets in the active workbook

WORKGROUP

Equivalent to clicking the Group Edit command on the Options menu in Microsoft Excel version 4.0. Creates a group. This function is provided for compatibility only. In Microsoft Excel version 5.0 and later versions, you can create a group by using the WORKBOOK.SELECT function.

Syntax

WORKGROUP(name_array)

WORKGROUP?(name_array)

Name_array is the list of workbooks or sheets in workbooks that you want grouped.

- If name_array is omitted, the most recently created group is recreated.
- If no group has been created during the current Microsoft Excel session, all open, unhidden worksheets are created as a group.
- If you specify just the name of a workbook, WORKGROUP adds the first sheet of the workbook to the group.

Remarks

WORKGROUP returns the #VALUE! error value and interrupts the macro if it can't find any of the sheets in name_array or if any of the sheets is a chart or module.

Related Functions

FILL.GROUP Fills the contents of the active worksheet's selection to the same area on all other worksheets in the group

WORKBOOK.SELECT Selects one or more sheets in a workbook

WORKSPACE

Changes the workspace settings for a workbook. This function is provide for compatibility with Microsoft Excel version 4.0 only. In Microsoft Excel version 5.0 and later versions, you can change workbook settings with OPTIONS.GENERAL function.

Syntax

WORKSPACE(fixed, decimals, r1c1, scroll, status, formula, menu_key, remote, entermove, underlines, tools, notes, nav_keys, menu_key_action, drag_drop, show_info)

WORKSPACE?(fixed, decimals, r1c1, scroll, status, formula, menu_key, remote, entermove, underlines, tools, notes, nav_keys, menu_key_action, drag_drop, show_info)

Arguments correspond to check boxes and text boxes in the Workspace dialog box. Arguments corresponding to check boxes are logical values. If an argument is TRUE, the check box is selected; if FALSE, the check box is cleared; if omitted, the current setting is not changed.

Fixed corresponds to the Fixed Decimal check box.

Decimals specifies the number of decimal places. Decimals is ignored if fixed is FALSE or omitted.

R1c1 corresponds to the R1C1 check box.

Scroll corresponds to the Scroll Bars check box.

Status   corresponds to the Status Bar check box.

Formula   corresponds to the Formula Bar check box.

Menu_key   is a text value indicating an alternate menu key, and corresponds to the Alternate Menu Or Help Key box.

Remote   corresponds to the Ignore Remote Requests check box.

Important   Microsoft Excel for the Macintosh requires system software version 7.0 or later for this argument.

Entermove   corresponds to the Move Selection After Enter/Return check box.

Underlines   is a number corresponding to the Command Underline options as shown in the following table.

Note   This argument is only available in Microsoft Excel for the Macintosh.

If underlines is	Command underlines are
1	On
2	Off
3	Automatic

Tools   is a logical value. If TRUE, the Standard toolbar is displayed; if FALSE, all visible toolbars are hidden. If omitted, the current toolbar display is not changed.

Notes   corresponds to the Note Indicator check box.

Nav_keys   corresponds to the Alternate Navigation Keys check box. In Microsoft Excel for the Macintosh, nav_keys is ignored.

Menu_key_action   is the number 1 or 2 specifying options for the alternate menu or Help key. In Microsoft Excel for the Macintosh, menu_key_action is ignored.

Menu_key_action	Alternate menu or Help key activates
1 or omitted	Microsoft Excel menus
2	Lotus 1-2-3 Help

Drag_drop   corresponds to the Cell Drag And Drop check box.

Show_info   corresponds to the Info Window check box.

Related Function

GET.WORKSPACE   Returns information about the workspace

ZOOM

Equivalent to choosing the Zoom command from the View menu. Enlarges or reduces a sheet in the active window. Use ZOOM when you need to view more cells than would normally fit in the active windows, or fewer cells at a larger size.

Syntax

ZOOM(magnification)

Magnification is a logical value or a number specifying the amount of enlargement or reduction.

- Magnification can be a number from 10 to 400 specifying the percentage of enlargement or reduction.
- If magnification is TRUE or omitted, the current selection is enlarged or reduced to completely fill the active window.
- If magnification is FALSE, the sheet is restored to normal 100% magnification.

Related Function

PRINT.PREVIEW Previews pages and page breaks before printing.

ZTESTM

Performs a two-sample z-test for means, assuming the two samples have known variances.

If this function is not available, you must install the Analysis ToolPak add-in.

Syntax

ZTESTM(**inprng1, inprng2**, outrng, labels, alpha, difference, **var1, var2**)

ZTESTM?(inprng1, inprng2, outrng, labels, alpha, difference, var1, var2)

Inprng1 is the input range for the first data set.

Inprng2 is the input range for the second data set.

Outrng is the first cell (the upper-left cell) in the output table or the name, as text, of a new sheet to contain the output table. If FALSE, blank, or omitted, places the output table in a new workbook.

Labels is a logical value.

- If labels is TRUE, then the first row or column of the input ranges contains labels.
- If labels is FALSE or omitted, all cells in inprng1 and inprng2 are considered data. Microsoft Excel will then generate the appropriate data labels for the output table.

Alpha is the confidence level for the test. If omitted, alpha is 0.05.

Difference is the hypothesized difference in means. If omitted, difference is 0.

Var1 is the variance of the first data set.

Var2 is the variance of the second data set.

myOnlineTraininghub

MORE RESOURCES

Weekly Newsletter

100 Tips & Tricks eBook & Newsletter	https://www.myonlinetraininghub.com/sign-up-for-100-excel-tips-and-tricks

Tutorials

Excel Formulas	https://www.myonlinetraininghub.com/excel-formulas
PivotTable Blog Posts	https://www.myonlinetraininghub.com/category/excel-pivottables
Power Query Blog Posts	https://www.myonlinetraininghub.com/category/power-query
Power Pivot Blog Posts	https://www.myonlinetraininghub.com/category/power-pivot
Excel Macros and VBA Blog Posts	https://www.myonlinetraininghub.com/category/excel-vba
Charting Blog Posts	http://www.myonlinetraininghub.com/category/excel-charts
Excel Dashboard Blog Posts	http://www.myonlinetraininghub.com/category/excel-dashboard

Free Webinars

Excel Dashboards	http://www.myonlinetraininghub.com/excel-webinars
Power BI	https://www.myonlinetraininghub.com/excel-dashboard-webinar-reg.htm#pbi_anchor

Support

Excel Forum	https://www.myonlinetraininghub.com/excel-forum

Courses

Complete Course Library	https://www.myonlinetraininghub.com/
Premium Bundle (Excel, Word & Outlook)	https://www.myonlinetraininghub.com/microsoft-office-online-training-courses
Advanced Excel	https://www.myonlinetraininghub.com/excel-expert-upgrade
PivotTables	https://www.myonlinetraininghub.com/excel-pivottable-course

Power Query	https://www.myonlinetraininghub.com/excel-power-query-course
Power Pivot	https://www.myonlinetraininghub.com/power-pivot-course
Excel for Decision Making	https://www.myonlinetraininghub.com/excel-for-decision-making-course
Excel for Customer Service	https://www.myonlinetraininghub.com/excel-for-customer-service-professionals
Excel for Finance	https://www.myonlinetraininghub.com/excel-for-finance-course
Excel Analysis ToolPak	https://www.myonlinetraininghub.com/excel-analysis-toolpak-course
Excel Dashboards	http://www.myonlinetraininghub.com/excel-dashboard-course
Power BI	http://www.myonlinetraininghub.com/power-bi-course

ABOUT

I've worked in many areas during my career; desktop support, server administration, network engineering and project management to name a few. One thing that I've always done is some sort of programming.

I started with BASIC and then moved to Pascal, SQL, C and assembler. I did a lot of DOS shell scripting. These days I find myself writing PHP pretty much every day along with HTML/CSS, JavaScript and of course, VBA.

When I get some spare time I'll be learning Python and R.

Please feel free to share this eBook with your friends and colleagues. The more people I can help, the better. If you have any feedback or suggestions, I'd love to hear from you.

Philip Treacy

THANKS

I'd like to thank and give credit to Ian Page who put together a number of separate documents listing these macros on this forum, which I used to put this book together.

http://www.excelforum.com/excel-formulas-and-functions/1170158-xl4-macro-functions.html