

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 choose the Cells command from 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.

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)
```

ELL

You can retrieve interesting information about worksheet cells by using the Excel4 macro function GET.CELL. Define the name HasFormula with the value =GET.CELL(48,INDIRECT("RC[-1]",FALSE)) for example. If you now insert =HasFormula next right to a cell, you will be shown whether the cell has a formula (True) or not (False).

Another example for GET.CELL you can find [here](#).

An overview over some arguments for GET.CELL:

| Proposed Name | Arg # | What =GET.CELL(Arg #,INDIRECT("RC[-1]",,)) will return |
|---------------|-------|--|
| AbsReference | 1 | Absolute style reference like [Book1.xls]Sheet1!\$A\$1 |

| | | |
|----------------|----|---|
| ShowValue | 5 | Cell value |
| ShowFormula | 6 | Cell formula |
| NumFormat | 7 | Number format of cell |
| IsLocked | 14 | True if cell is locked |
| FormulaHidden | 15 | True if cell formula is hidden |
| ShowWidth | 16 | Cell width. If array-entered into two cells of a row, second value is true if width is standard |
| ShowHeight | 17 | Cell height |
| WorkbookName | 32 | Workbook name like [Book1.xls]Sheet1 or Book1.xls if workbook and single sheet have identical names |
| ShowFormulaWOT | 41 | Cell formula without translation into language of workspace |
| HasNote | 46 | True if cell has a text note |
| HasFormula | 48 | True if cell contains a formula |
| IsArray | 49 | True if cell is part of an array formula |
| IsStringConst | 52 | Text alignment char ' if cell is a string constant, empty string if not |
| AsText | 53 | Cell displayed as text with numbers formatted and symbols included |
| WorksheetName | 62 | Worksheet name like [Book1.xls]Sheet1 |
| WorkbookName | 66 | Workbook name like Book1.xls |
| IsHidden | | VBA only: True if cell is hidden (the entire row or column, actually) |

If you want to achieve similar results with VBA use this UDF:

```
Function sbGetCell(r As Range, s As String) As Variant
'Reverse("moc.LiborPlus.www") V0.11 PB 29-Jan-2011
Application.Volatile
Select Case s
Case "AbsReference", "1"
'Absolute style reference like [Book1.xls]Sheet1!$A$1
If Application.Caller.Parent.Parent.Name = r.Worksheet.Parent.Name And _
Application.Caller.Parent.Name = r.Worksheet.Name Then
sbGetCell = r.Address
Else
If InStr(r.Worksheet.Parent.Name & r.Worksheet.Name, " ") > 0 Then
```

```

        sbGetCell = "[" & r.Worksheet.Parent.Name & "]" & r.Worksheet.Name & "!" &
r.Address
    Else
        sbGetCell = "[" & r.Worksheet.Parent.Name & "]" & r.Worksheet.Name & "!" &
r.Address
    End If
End If
Case "ShowValue", "5"
    'Cell value
    sbGetCell = r.Value
Case "ShowFormula", "6"
    'Cell formula
    sbGetCell = r.FormulaLocal
Case "NumFormat", "7"
    'Number format of cell
    sbGetCell = r.NumberFormatLocal
Case "IsLocked", "14"
    'True if cell is locked
    sbGetCell = r.Locked
Case "FormulaHidden", "15"
    'True if cell formula is hidden
    sbGetCell = r.FormulaHidden
Case "ShowWidth", "16"
    'Cell width. If array-entered into two cells of a row, second value is true if width is
standard
    sbGetCell = r.ColumnWidth 'Not width!
Case "ShowHeight", "17"
    'Cell height
    sbGetCell = r.RowHeight
Case "WorkbookSheetName", "32"
    'Workbook name like [Book1.xls]Sheet1 or Book1.xls if workbook and single sheet have
    'identical names
    If r.Worksheet.Parent.Name = r.Worksheet.Name & ".xls" And _
        Application.Worksheets.Count = 1 Then
        sbGetCell = r.Worksheet.Parent.Name
    Else
        sbGetCell = "[" & r.Worksheet.Parent.Name & "]" & r.Worksheet.Name
    End If
Case "ShowFormulaWOT", "41"
    'Cell formula without translation into language of workspace
    sbGetCell = r.Formula
Case "HasNote", "46"
    'True if cell has a text note
    sbGetCell = Len(r.NoteText) > 0
Case "HasFormula", "48"
    'True if cell contains a formula
    sbGetCell = r.HasFormula
Case "IsArray", "49"
    'True if cell is part of an array formula
    sbGetCell = r.HasArray
Case "IsStringConst", "52"
    'Text alignment char "" if cell is a string constant, empty string "" if not
    sbGetCell = r.PrefixCharacter

```

```
Case "AsText", "53"  
    'Cell displayed as text with numbers formatted and symbols included  
    sbGetCell = Format(r.Value, r.NumberFormatLocal)  
Case "WorksheetName", "62"  
    'Worksheet name like [Book1.xls]Sheet1  
    sbGetCell = "[" & r.Worksheet.Parent.Name & "]" & r.Worksheet.Name  
Case "WorkbookName", "66"  
    'Workbook name like Book1.xls  
    sbGetCell = r.Worksheet.Parent.Name  
Case "IsHidden"  
    'Cell hidden?  
    sbGetCell = r.EntireRow.Hidden Or r.EntireColumn.Hidden  
Case Else  
    sbGetCell = CVErr(xlErrValue)  
End Select  
  
End Function
```

