Microsoft

# Microsoft Excel 2019

## Formulas and Functions

Paul McFedries

Sample files
on the web

Microsoft

# Microsoft Excel 2019 Formulas and Functions

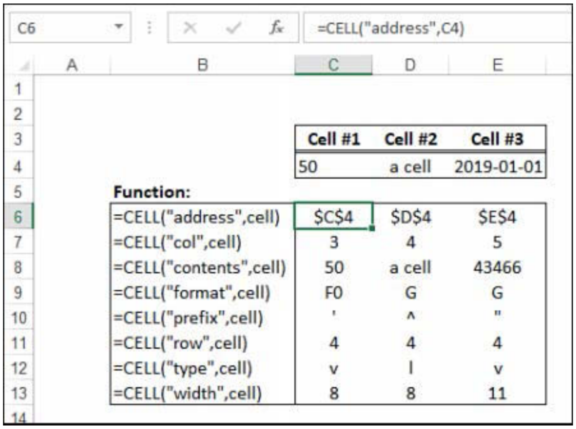
Paul McFedries

**TABLE 6-3** The CELL() function's info_type argument

| Info_Type Value | What CELL() Returns | |
|---|---|---|
| **address** | The absolute address, as text, of the *reference* cell. | |
| **col** | The column number of *reference*. | |
| **color** | Returns **1** if *reference* has a custom cell format that displays negative values in a color; returns **0** otherwise. | |
| **contents** | The contents of *reference*. | |
| **filename** | The full path and file name of the file that contains *reference*, as text. Returns the null string (**""**) if the workbook that contains *reference* hasn't been saved for the first time. | |
| **format** | A string that corresponds to the built-in Excel numeric format applied to *reference*. Here are the possible return values: | |
| | **Built-In Format** CELL() **Returns** | |
| | **General** | **G** |
| | **0** | **F0** |
| | **#,##0** | **,0** |
| | **0.00** | **F2** |
| | **#,##0.00** | **,2** |
| | **$#,##0_);($#,##0)** | **C0** |
| | **$#,##0_);[Red]($#,##0)** | **C0-** |
| | **$#,##0.00_);($#,##0.00)** | **C2** |
| | **$#,##0.00_);[Red]($#,##0.00)** | **C2-** |
| | **0%** | **P0** |
| | **0.00%** | **P2** |
| | **0.00E+00** | **S2** |
| | **# ?/?** or **# ??/??** | **G** |
| | **d-mmm-yy** or **dd-mmm-yy** | **D1** |
| | **d-mmm** or **dd-mmm** | **D2** |
| | **mmm-yy** | **D3** |
| | **m/d/yy** or **m/d/yy h:mm** or **mm/dd/yy** | **D4** |
| | **mm/dd** | **D5** |
| | **h:mm:ss AM/PM** | **D6** |
| | **h:mm AM/PM** | **D7** |
| | **h:mm:ss** | **D8** |
| | **h:mm** | **D9** |

| | | |
|---|---|---|
| **parentheses** | Returns **1** if *reference* has a custom cell format that uses parentheses for positive or all values; returns **0** otherwise. | |
| **prefix** | A character that represents the text alignment used by *reference*. Here are the possible return values: | |
| | **Alignment** | **CELL() Returns** |
| | Left | ' |
| | Center | ^ |
| | Right | " |
| | Fill | \ |
| **protect** | Returns **0** if *reference* isn't locked; **1** otherwise. | |
| **row** | The row number of *reference*. | |
| **type** | A letter that represents the type of data in the *reference*. Here are the possible return values: | |
| | **Data Type** | **CELL() Returns** |
| | Text | **L** |
| | Blank | **B** |
| | All others | **V** |
| **width** | The column width of *reference*, rounded to the nearest integer, where one unit equals the width of one character in the default font size. | |

Figure 6-14 shows how the **CELL()** function works.



C6 =CELL("address",C4)

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | | | | | |
| 2 | | | | | |
| 3 | | | **Cell #1** | **Cell #2** | **Cell #3** |
| 4 | | | 50 | a cell | 2019-01-01 |
| 5 | | **Function:** | | | |
| 6 | | =CELL("address",cell) | $C$4 | $D$4 | $E$4 |
| 7 | | =CELL("col",cell) | 3 | 4 | 5 |
| 8 | | =CELL("contents",cell) | 50 | a cell | 43466 |
| 9 | | =CELL("format",cell) | F0 | G | G |
| 10 | | =CELL("prefix",cell) | ' | ^ | " |
| 11 | | =CELL("row",cell) | 4 | 4 | 4 |
| 12 | | =CELL("type",cell) | v | l | v |
| 13 | | =CELL("width",cell) | 8 | 8 | 11 |
| 14 | | | | | |

**FIGURE 6-14** These are examples of the CELL() function.

## The **ERROR.TYPE()** function

The **ERROR.TYPE()** function returns a value that corresponds to a specific Excel error value:

```
ERROR.TYPE(error_val)
```

*error_val* A reference to a cell containing a formula that you want to check for the error value. Here are the possible return values:

| | *error_val* **Value** | **ERROR.TYPE() Returns** |
|---|---|---|
| | **#NULL!** | **1** |
| | **#DIV/0!** | **2** |
| | **#VALUE!** | **3** |
| | **#REF!** | **4** |
| | **#NAME?** | **5** |
| | **#NUM!** | **6** |
| | **#N/A!** | **7** |
| | **#GETTING_DATA** | **8** |
| | **#SPILL!** | 9 |
| | **#UNKNOWN!** | 12 |
| | **#FIELD!** | 13 |
| | **#CALC!** | **14** |
| | All others | **#N/A** |

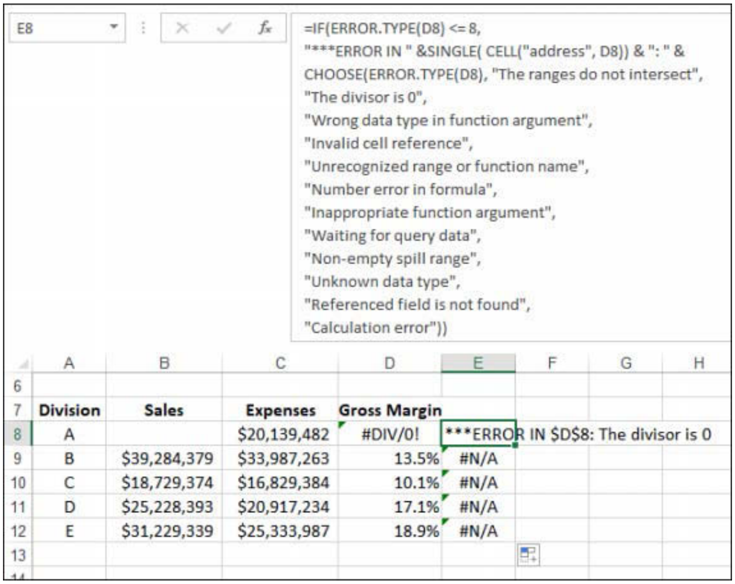
You most often use the **ERROR.TYPE()** function to intercept an error and then display a more useful or friendly message. You do this by using the **IF()** function to see if **ERROR.TYPE()** returns a value less than or equal to **7**; if so, the cell in question contains an error value. Because the **ERROR.TYPE()** returns value ranges from **1** to **8**, you can apply the return value to the **CHOOSE()** function to display the error message.

For the details of the **CHOOSE()** function, **see** "The **CHOOSE()** function," in Chapter 7.

Here's a formula that does all that. (I've split the formula so that different parts appear on different lines to make it easier for you to see what's going on.)

```
=IF(ERROR.TYPE(D8) <= 8,
    ***ERROR IN " & CELL("address",D8) & ": " &
    CHOOSE(ERROR.TYPE(D8),"The ranges do not intersect",
    "The divisor is 0",
    "Wrong data type in function argument",
    "Invalid cell reference",
    "Unrecognized range or function name",
    "Number error in formula",
    "Inappropriate function argument",
    "Waiting for query data",
    "Non-empty spill range",
    "Unknown data type",
    "Referenced field is not found",
    "Calculation error"))
```

Figure 6-15 shows this formula in an example. (Note that the formula displays **#N/A** when there is no error; this is the return value of **ERROR.TYPE()** when there is no error.)



```
=IF(ERROR.TYPE(D8) <= 8,
"***ERROR IN " &SINGLE( CELL("address", D8)) & ": " &
CHOOSE(ERROR.TYPE(D8), "The ranges do not intersect",
"The divisor is 0",
"Wrong data type in function argument",
"Invalid cell reference",
"Unrecognized range or function name",
"Number error in formula",
"Inappropriate function argument",
"Waiting for query data",
"Non-empty spill range",
"Unknown data type",
"Referenced field is not found",
"Calculation error"))
```

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 6 | | | | | | | | |
| 7 | **Division** | **Sales** | **Expenses** | **Gross Margin** | | | | |
| 8 | A | | $20,139,482 | #DIV/0! | ***ERROR IN $D$8: The divisor is 0 | | | |
| 9 | B | $39,284,379 | $33,987,263 | 13.5% | #N/A | | | |
| 10 | C | $18,729,374 | $16,829,384 | 10.1% | #N/A | | | |
| 11 | D | $25,228,393 | $20,917,234 | 17.1% | #N/A | | | |
| 12 | E | $31,229,339 | $25,333,987 | 18.9% | #N/A | | | |
| 13 | | | | | | | | |

**FIGURE 6-15** This formula uses IF() and ERROR_TYPE() to return a more descriptive error message to the user.

## The INFO() function

The **INFO()** function is seldom used, but it's handy when you need it because it gives you information about the current operating environment:

```
INFO(type_text)
```

*type_text* A string that specifies the type of information you want.

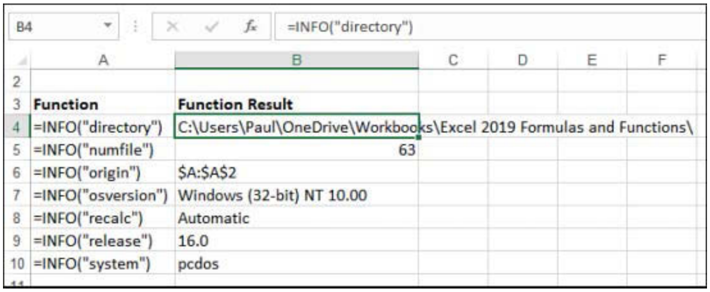
Table 6-4 lists the possible values for the *type_text* argument.

**TABLE 6-4** The INFO() function's type_text argument

| ***type_text* Value** | **What INFO() Returns** |
|---|---|
| **directory** | The full pathname of the current folder. (That is, the folder that will appear the next time you display the Open or Save As dialog box.) |
| **numfile** | The number of worksheets in all the open workbooks, including hidden ones. |
| **origin** | The address of the upper-left cell that is visible in the current worksheet. In Figure 6-16, for example, cell A2 is the visible cell in the upper-left corner. The absolute address begins with **$A:** for Lotus 1-2-3 release 3.x compatibility. |

| | |
|---|---|
| **osversion** | A string containing the current operating system version. |
| **recalc** | A string containing the current recalculation mode: **Automatic** or **Manual**. |
| **release** | A string containing the version of Microsoft Excel. |
| **system** | A string containing a code representing the current operating environment: **pcdos** for Windows or **mac** for Macintosh. |

Figure 6-16 shows the **INFO()** function at work.



B4 =INFO("directory")

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 2 | | | | | | |
| 3 | **Function** | **Function Result** | | | | |
| 4 | =INFO("directory") | C:\Users\Paul\OneDrive\Workbooks\Excel 2019 Formulas and Functions\ | | | | |
| 5 | =INFO("numfile") | 63 | | | | |
| 6 | =INFO("origin") | $A:$A$2 | | | | |
| 7 | =INFO("osversion") | Windows (32-bit) NT 10.00 | | | | |
| 8 | =INFO("recalc") | Automatic | | | | |
| 9 | =INFO("release") | 16.0 | | | | |
| 10 | =INFO("system") | pcdos | | | | |

**FIGURE 6-16** The INFO() function is in action here.

## The SHEET() and SHEETS() functions

Excel includes two information functions—**SHEET()** and **SHEETS()**—that return information about the worksheets in a workbook. You use the **SHEET()** function to return a sheet number using the following syntax:

```
SHEET([value])
```

*value* An optional value that specifies a sheet. If you omit *value*, Excel references the current sheet.

For example, the formula **=SHEET()** returns the number of the sheet that contains the formula, where **1** is the first sheet in the workbook, **2** is the second sheet, and so on. Note that Excel counts all sheet types, including worksheets and chart sheets.

If a worksheet has the name **Budget**, then the formula **=SHEET("Budget")** returns its sheet number. Alternatively, you can use a cell reference within that sheet, such as **SHEET(Budget!A1)**.

You use the **SHEETS()** function (which takes no arguments) to return the total number of worksheets in the current workbook.

## The IS functions

Excel's so-called **IS** functions are Boolean functions that return either **TRUE** or **FALSE**, depending on the argument they're evaluating:

```
ISBLANK(value)
ISERR(value)
ISERROR(value)
ISEVEN(number)
ISFORMULA(reference)
ISLOGICAL(value)
ISNA(value)
ISNONTEXT(value)
ISNUMBER(value)
ISODD(number)
ISREF(value)
ISTEXT(value)
```

| | |
|---|---|
| *value* | A cell reference, function return value, or formula result |
| *reference* | A cell reference |
| *number* | A numeric value |

The operation of these functions is straightforward, so rather than run through the specifics of all 11 functions, in the next few sections I show you some interesting and useful techniques that make use of these functions.

## Counting the number of blanks in a range

When putting together the data for a worksheet model, it's common to pull the data from various sources. Unfortunately, this often means that the data arrives at different times, and you end up with an incomplete model. If you're working with a big list, you might want to keep a running total of the number of pieces of data you're still missing.

This is the perfect opportunity to break out the **ISBLANK()** function and plug it into the array formula for counting that you learned earlier:

```
=SUM(IF(ISBLANK(range), 1, 0))
```

The **IF()** function runs through the *range*, looking for blank cells. Each time it comes across a blank cell, it returns **1**; otherwise, it returns **0**. The **SUM()** function adds the results to give the total number of blank cells. Figure 6-17 shows an example (see cell G1).

**Tip** Using an array formula to count blank cells is fine, but it's not the easiest way to go about it. In most cases, you're better off just using the **COUNTBLANK(*range*)** function, which counts the number of blank cells that occur in the range specified by the *range* argument.