Microsoft

Microsoft Excel Formulas and Functions

(Office 2021 and Microsoft 365)

Paul McFedries

Sample files on the web

Microsoft

Microsoft Excel Formulas and Functions (Office 2021 and Microsoft 365)

Paul McFedries

Counting occurrences in a range

Now you know how to find out whether a value appears in a list, but what if you need to know how many times the value appears? The following formula does the job:

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

Again, *value* is the value you want to look up, and *range* is the range for searching. In this array formula, the **IF** function compares *value* with every cell in *range*. The values that match return **1**, and those that don't return **0**. The **SUM** function adds these returned values, and the final total is the number of occurrences of *value*. Here's a formula that does this for our list of overdue invoices:

```
=SUM(IF(B1 = B6:B29, 1, 0))
```

Figure 6-9 shows this formula in action (see cell B3).

B3 {=SUM(IF(B1 = B6:B29, 1, 0))}

	A	B	C	D	E	F	G
1	**Account Number:**	09-2111					
2	**In the List?**	TRUE	**First Row:**	14			
3	**How Many Times?**	2	**Last Row:**	22			
4							
5	**Account Name**	**Account Number**	**Invoice Number**	**Invoice Amount**	**Due Date**	**Date Paid**	**Days Overdue**
6	Emily's Sports Palace	08-2255	117316	$ 1,584.20	26-Apr-19		55
7	Refco Office Solutions	14-5741	117317	$ 303.65	27-Apr-19		54
8	Brimson Furniture	10-0009	117321	$ 2,144.55	3-May-19		48
9	Katy's Paper Products	12-1212	117322	$ 234.69	4-May-19		47
10	Door Stoppers Ltd.	01-0045	117324	$ 101.01	10-May-19		41
11	Voyatzis Designs	14-1882	117325	$ 1,985.25	10-May-19		41
12	Brimson Furniture	10-0009	117327	$ 1,847.25	16-May-19		35
13	Door Stoppers Ltd.	01-0045	117328	$ 58.50	17-May-19		34
14	O'Donoghue Inc.	09-2111	117329	$ 1,234.56	18-May-19		33
15	Refco Office Solutions	14-5741	117330	$ 456.78	18-May-19		33
16	Renaud & Son	07-0025	117331	$ 565.77	23-May-19		30
17	Simpson's Ltd.	16-6658	117332	$ 898.54	22-May-19		31
18	Door Stoppers Ltd.	01-0045	117333	$ 1,685.74	26-May-19		27
19	Renaud & Son	07-0025	117335	$ 3,005.14	28-May-19		25
20	Rooter Office Solvents	07-4441	117336	$ 78.85	30-May-19		23
21	Emily's Sports Palace	08-2255	117337	$ 4,347.21	2-Jun-19		20
22	O'Donoghue Inc.	09-2111	117338	$ 2,144.55	2-Jun-19		20
23	Brimson Furniture	10-0009	117339	$ 1,234.69	3-Jun-19		19

FIGURE 6-9 This worksheet uses SUM and IF in an array formula to count the number of occurrences of a value in a list.

Note The generic array formula **=SUM(IF(*condition*, 1, 0))** is useful in any context where you need to count the number of occurrences in which *condition* returns **TRUE**. The *condition* argument is normally a logical formula that compares a single value with each cell in a range of values. However, it's also possible to compare two ranges, as long as they're the same shape. (That is, they have the same number of rows and columns.) For example, suppose that you want to compare the values in two ranges named **Range1** and **Range2** to see if any of the values are different. Here's an array formula that does this:

```
=SUM(IF(Range1 <> Range2, 1, 0))
```

This formula compares the first cell in **Range1** with the first cell in **Range2**, the second cell in **Range1** with the second cell in **Range2**, and so on. Each time the values don't match, the comparison returns **1**; otherwise, it returns **0**. The sum of these comparisons is the number of different values between the two ranges.

Determining where a value appears in a list

What if you want to know not just whether a value appears in a list but *where* it appears in the list? You can do this by getting the **IF** function to return the row number for a positive result:

```
IF(value = range, ROW(range), "")
```

Whenever *value* equals one of the cells in *range*, the **IF** function uses **ROW** to return the row number; otherwise, it returns the empty string.

To return that row number, use either the **MIN** function or the **MAX** function, which returns the minimum or maximum, respectively, in a collection of values. The trick here is that both functions ignore null values, so applying this to the array that results from the previous **IF** expression tells where the matching values are:

- To get the first instance of the value, use the **MIN** function in an array formula, like so:

  ```
  =MIN(IF(value = range, ROW(range), ""))
  ```

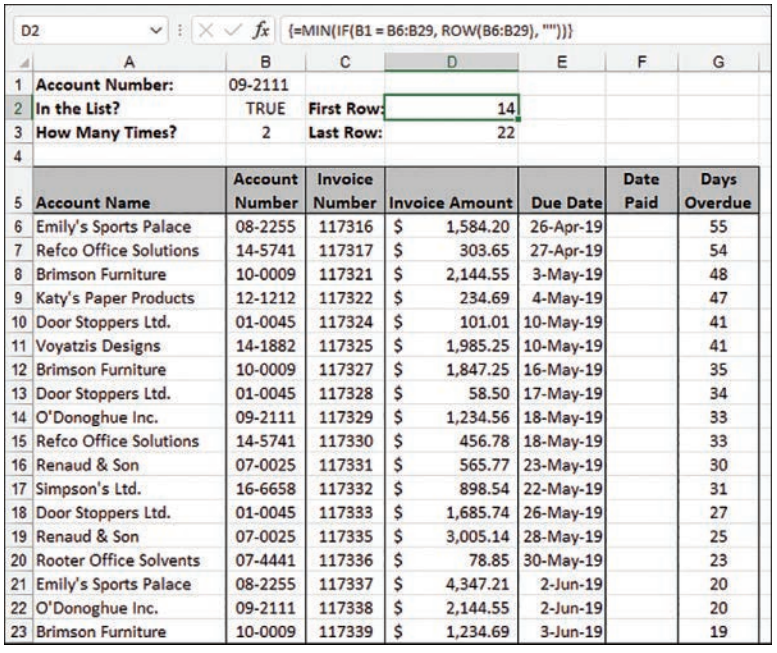
- To get the last instance of the value, use the **MAX** function in an array formula, as shown here:

  ```
  =MAX(IF(value = range, ROW(range), ""))
  ```

Here are the formulas you would use to find the first and last occurrences in the previous list of overdue invoices:

```
=MIN(IF(B1 = B6:B29, ROW(B6:B29), ""))
=MAX(IF(B1 = B6:B29, ROW(B6:B29), ""))
```

Figure 6-10 shows the results (with the row of the first occurrence in cell D2 and the row of the last occurrence in cell D3).



D2 {=MIN(IF(B1 = B6:B29, ROW(B6:B29), ""))}

	A	B	C	D	E	F	G
1	**Account Number:**	09-2111					
2	**In the List?**	TRUE	**First Row:**	14			
3	**How Many Times?**	2	**Last Row:**	22			
4							
5	**Account Name**	**Account Number**	**Invoice Number**	**Invoice Amount**	**Due Date**	**Date Paid**	**Days Overdue**
6	Emily's Sports Palace	08-2255	117316	$ 1,584.20	26-Apr-19		55
7	Refco Office Solutions	14-5741	117317	$ 303.65	27-Apr-19		54
8	Brimson Furniture	10-0009	117321	$ 2,144.55	3-May-19		48
9	Katy's Paper Products	12-1212	117322	$ 234.69	4-May-19		47
10	Door Stoppers Ltd.	01-0045	117324	$ 101.01	10-May-19		41
11	Voyatzis Designs	14-1882	117325	$ 1,985.25	10-May-19		41
12	Brimson Furniture	10-0009	117327	$ 1,847.25	16-May-19		35
13	Door Stoppers Ltd.	01-0045	117328	$ 58.50	17-May-19		34
14	O'Donoghue Inc.	09-2111	117329	$ 1,234.56	18-May-19		33
15	Refco Office Solutions	14-5741	117330	$ 456.78	18-May-19		33
16	Renaud & Son	07-0025	117331	$ 565.77	23-May-19		30
17	Simpson's Ltd.	16-6658	117332	$ 898.54	22-May-19		31
18	Door Stoppers Ltd.	01-0045	117333	$ 1,685.74	26-May-19		27
19	Renaud & Son	07-0025	117335	$ 3,005.14	28-May-19		25
20	Rooter Office Solvents	07-4441	117336	$ 78.85	30-May-19		23
21	Emily's Sports Palace	08-2255	117337	$ 4,347.21	2-Jun-19		20
22	O'Donoghue Inc.	09-2111	117338	$ 2,144.55	2-Jun-19		20
23	Brimson Furniture	10-0009	117339	$ 1,234.69	3-Jun-19		19

FIGURE 6-10 This worksheet uses MIN, MAX, ROW, and IF in array formulas to return the row numbers of the first (cell D2) and last (cell D3) occurrences of a value in a list.

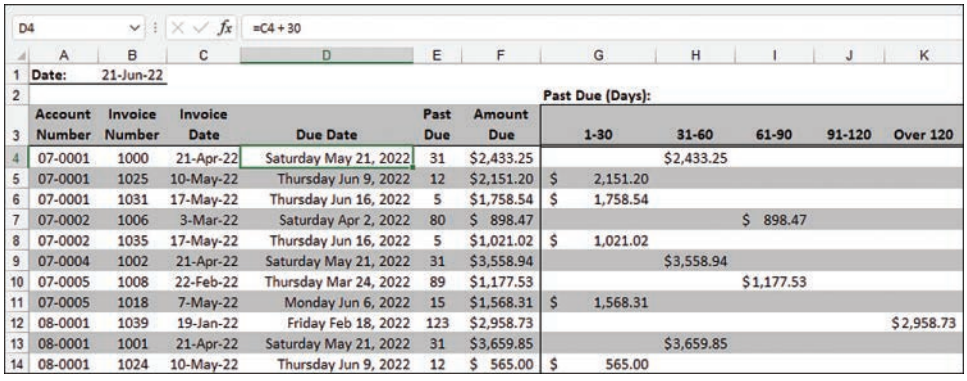
Tip It's also possible to determine the address of the cell that contains the first or last occurrence of a value in a list. To do this, use the **ADDRESS** function, which returns an absolute address, given a row and column number:

```
=ADDRESS(MIN(IF(B1 = B6:B29, ROW(B6:B29), "")), COLUMN(B6:B29))
=ADDRESS(MAX(IF(B1 = B6:B29, ROW(B6:B29), "")), COLUMN(B6:B29))
```

Case study: Building an accounts receivable aging worksheet

If you use Excel to store accounts receivable data, it's a good idea to set up an aging worksheet that shows past-due invoices, calculates the number of days past due, and groups the invoices into past-due categories (1–30 days, 31–60 days, and so on).

Figure 6-11 shows a simple implementation of an accounts receivable database. For each invoice, the due date (column D) is calculated by adding 30 to the invoice date (column C). Column E subtracts the due date (column D) from the current date (in cell B1) to calculate the number of days each invoice is past due.



D4 =C4 + 30

	A	B	C	D	E	F	G	H	I	J	K
1	Date:	21-Jun-22									
2							Past Due (Days):				
3	Account Number	Invoice Number	Invoice Date	Due Date	Past Due	Amount Due	1-30	31-60	61-90	91-120	Over 120
4	07-0001	1000	21-Apr-22	Saturday May 21, 2022	31	$2,433.25		$2,433.25			
5	07-0001	1025	10-May-22	Thursday Jun 9, 2022	12	$2,151.20	$ 2,151.20				
6	07-0001	1031	17-May-22	Thursday Jun 16, 2022	5	$1,758.54	$ 1,758.54				
7	07-0002	1006	3-Mar-22	Saturday Apr 2, 2022	80	$ 898.47			$ 898.47		
8	07-0002	1035	17-May-22	Thursday Jun 16, 2022	5	$1,021.02	$ 1,021.02				
9	07-0004	1002	21-Apr-22	Saturday May 21, 2022	31	$3,558.94		$3,558.94			
10	07-0005	1008	22-Feb-22	Thursday Mar 24, 2022	89	$1,177.53			$1,177.53		
11	07-0005	1018	7-May-22	Monday Jun 6, 2022	15	$1,568.31	$ 1,568.31				
12	08-0001	1039	19-Jan-22	Friday Feb 18, 2022	123	$2,958.73					$2,958.73
13	08-0001	1001	21-Apr-22	Saturday May 21, 2022	31	$3,659.85		$3,659.85			
14	08-0001	1024	10-May-22	Thursday Jun 9, 2022	12	$ 565.00	$ 565.00				

FIGURE 6-11 This is a simple accounts receivable database.

Calculating a smarter due date

You might have noticed a problem with the due dates in Figure 6-11: Several of the dates, including the date in cell D4, fall on weekends. The problem here is that the due date calculation just adds 30 to the invoice date. To avoid weekend due dates, you need to test whether the invoice date plus 30 falls on a Saturday or Sunday. The **WEEKDAY** function helps because it returns **7** if the date is a Saturday and **1** if the date is a Sunday.

So, to check for a Saturday, you could use the following formula:

```
=IF(WEEKDAY(C4 + 30) = 7, C4 + 32, C4 + 30)
```

Here, I'm assuming that the invoice date resides in cell C4. If **WEEKDAY(C4 + 30)** returns **7**, the date is a Saturday, so you add 32 to C4 instead (to make the due date the following Monday). Otherwise, you just add 30 days as usual.

Checking for a Sunday is similar:

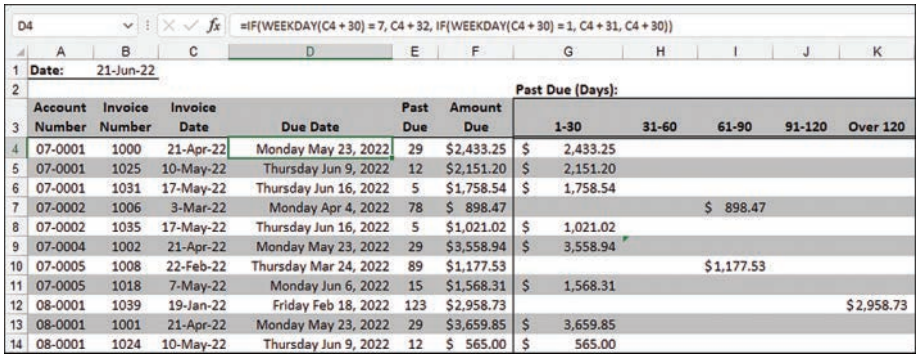
```
=IF(WEEKDAY(C4 + 30) = 1, C4 + 31, C4 + 30)
```

The problem, though, is that you need to combine these two tests into a single formula. To do that, you can nest one **IF** function inside another. Here's how it works:

```
=IF(WEEKDAY(C4+30) = 7, C4+32, IF(WEEKDAY(C4+30) = 1, C4+31, C4+30))
```

The main **IF** checks whether the date is a Saturday. If it is, you add 32 days to C4; otherwise, the formula runs the second **IF**, which checks for Sunday. Figure 6-12 shows the revised aging sheet with the nonweekend due dates in column D.

For calculating due dates based on workdays (that is, excluding weekends and holidays), Excel has a function named **WORKDAY** that handles this calculation with ease; see "A workday alternative: The **WORKDAY** function," in Chapter 8, "Working with date and time functions."



D4 =IF(WEEKDAY(C4 + 30) = 7, C4 + 32, IF(WEEKDAY(C4 + 30) = 1, C4 + 31, C4 + 30))

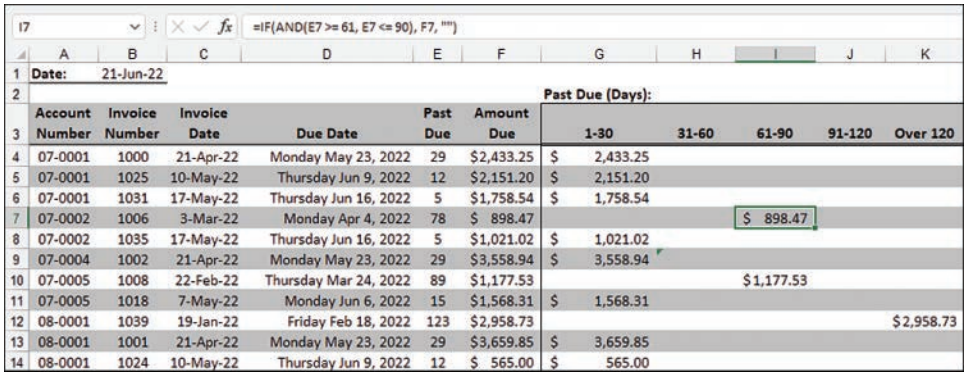
	A	B	C	D	E	F	G	H	I	J	K
1	Date:	21-Jun-22									
2							Past Due (Days):				
3	Account Number	Invoice Number	Invoice Date	Due Date	Past Due	Amount Due	1-30	31-60	61-90	91-120	Over 120
4	07-0001	1000	21-Apr-22	Monday May 23, 2022	29	$2,433.25	$ 2,433.25				
5	07-0001	1025	10-May-22	Thursday Jun 9, 2022	12	$2,151.20	$ 2,151.20				
6	07-0001	1031	17-May-22	Thursday Jun 16, 2022	5	$1,758.54	$ 1,758.54				
7	07-0002	1006	3-Mar-22	Monday Apr 4, 2022	78	$ 898.47			$ 898.47		
8	07-0002	1035	17-May-22	Thursday Jun 16, 2022	5	$1,021.02	$ 1,021.02				
9	07-0004	1002	21-Apr-22	Monday May 23, 2022	29	$3,558.94	$ 3,558.94				
10	07-0005	1008	22-Feb-22	Thursday Mar 24, 2022	89	$1,177.53			$1,177.53		
11	07-0005	1018	7-May-22	Monday Jun 6, 2022	15	$1,568.31	$ 1,568.31				
12	08-0001	1039	19-Jan-22	Friday Feb 18, 2022	123	$2,958.73					$2,958.73
13	08-0001	1001	21-Apr-22	Monday May 23, 2022	29	$3,659.85	$ 3,659.85				
14	08-0001	1024	10-May-22	Thursday Jun 9, 2022	12	$ 565.00	$ 565.00				

FIGURE 6-12 The revised worksheet uses the IF and WEEKDAY functions to ensure that due dates don't fall on weekends.

Aging overdue invoices

For cash-flow purposes, you also need to correlate the invoice amounts with the number of days past due. Ideally, you'd like to see a list of invoice amounts that are between 1 and 30 days past due, between 31 and 60 days past due, and so on. Figure 6-13 shows one way to set up accounts receivable aging.

The worksheet in Figures 6-11 through 6-13 uses ledger shading for easier reading. To learn how to apply ledger shading automatically, see "Creating ledger shading," in Chapter 9, "Working with math functions."



I7 =IF(AND(E7 >= 61, E7 <= 90), F7, "")

	A	B	C	D	E	F	G	H	I	J	K
1	Date:	21-Jun-22									
2							Past Due (Days):				
3	Account Number	Invoice Number	Invoice Date	Due Date	Past Due	Amount Due	1-30	31-60	61-90	91-120	Over 120
4	07-0001	1000	21-Apr-22	Monday May 23, 2022	29	$2,433.25	$ 2,433.25				
5	07-0001	1025	10-May-22	Thursday Jun 9, 2022	12	$2,151.20	$ 2,151.20				
6	07-0001	1031	17-May-22	Thursday Jun 16, 2022	5	$1,758.54	$ 1,758.54				
7	07-0002	1006	3-Mar-22	Monday Apr 4, 2022	78	$ 898.47			$ 898.47		
8	07-0002	1035	17-May-22	Thursday Jun 16, 2022	5	$1,021.02	$ 1,021.02				
9	07-0004	1002	21-Apr-22	Monday May 23, 2022	29	$3,558.94	$ 3,558.94				
10	07-0005	1008	22-Feb-22	Thursday Mar 24, 2022	89	$1,177.53			$1,177.53		
11	07-0005	1018	7-May-22	Monday Jun 6, 2022	15	$1,568.31	$ 1,568.31				
12	08-0001	1039	19-Jan-22	Friday Feb 18, 2022	123	$2,958.73					$2,958.73
13	08-0001	1001	21-Apr-22	Monday May 23, 2022	29	$3,659.85	$ 3,659.85				
14	08-0001	1024	10-May-22	Thursday Jun 9, 2022	12	$ 565.00	$ 565.00				

FIGURE 6-13 You can use IF and AND to categorize past-due invoices for aging purposes.

The aging worksheet calculates the number of days past due by subtracting the due date from the date shown in cell B1. If you calculate days past due using only workdays (weekends and holidays excluded), a better choice is the **NETWORKDAYS** function, covered in "**NETWORKDAYS**: Calculating the number of workdays between two dates," in Chapter 8.

For the invoice amounts shown in column G (1–30 days), the sheet uses the following formula (which appears in G4):

```
=IF(E4 <= 30, F4, "")
```

If the number of days the invoice is past due (cell E4) is less than or equal to 30, the formula displays the amount (from cell F4); otherwise, it displays a blank.

The amounts in column H (31–60 days) are a little trickier. Here, you need to check whether the number of days past due is greater than or equal to 31 days *and* less than or equal to 60 days. To accomplish this, you can press the **AND** function into service:

```
=IF(AND(E4 >= 31, E4 <= 60), F4, "")
```

The **AND** function checks two logical expressions: **E4> = 31** and **E4 <= 60**. If both are true, **AND** returns **TRUE**, and the **IF** function displays the invoice amount. If one of the logical expressions isn't true (or if they're both not true), **AND** returns **FALSE**, and the **IF** function returns a blank. Similar formulas appear in column I (61–90 days) and column J (91–120 days). Column K (Over 120) looks for past-due values that are greater than 120.

Getting data with information functions

Excel's information functions return data concerning cells, worksheets, and formula results. Table 6-2 lists all the information functions.

TABLE 6-2 Excel's information functions

Function	Description
CELL(*info_type***[,***reference***])**	Returns information about various cell attributes, including formatting, contents, and location.
ERROR.TYPE(*error_val***)**	Returns a number corresponding to an error type.
INFO(*type_text***)**	Returns information about the operating system and environment.
ISBLANK(*value***)**	Returns **TRUE** if *value* is blank.
ISERR(*value***)**	Returns **TRUE** if *value* is any error value except **#N/A**.
ISERROR(*value***)**	Returns **TRUE** if *value* is any error value.
ISEVEN(*number***)**	Returns **TRUE** if *number* is even.
ISFORMULA(*reference***)**	Returns **TRUE** if the cell specified by *reference* contains a formula.
ISLOGICAL(*value***)**	Returns **TRUE** if *value* is a logical value.
ISNA(*value***)**	Returns **TRUE** if *value* is the **#N/A** error value.
ISNONTEXT(*value***)**	Returns **TRUE** if *value* is not text.
ISNUMBER(*value***)**	Returns **TRUE** if *value* is a number.
ISODD(*number***)**	Returns **TRUE** if *number* is odd.
ISOMITTED(*argument***)**	Returns **TRUE** if the *argument* parameter is missing from a LAMBDA function (see Chapter 9).
ISREF(*value***)**	Returns **TRUE** if *value* is a reference.
ISTEXT(*value***)**	Returns **TRUE** if *value* is text.