

# Finding the last cell in an Excel range

*Using built in Excel functions to locate the last cell containing data in a range of cells.*

---

There are all sorts of times in Excel when you will need to find the last cell in a range. For instance, in using a dynamic range name, it is vital to be able to detect which cells actually contain data. This article will explain and evaluate the different approaches.

Let's use the data below as our starting point.

	A	B	C
1	<b>Name</b>	<b>Score</b>	<b>Region</b>
2	Andy	90	S
3	Bill	80	S
4	Drew	60	N
5	Charlie	90	
6		50	N
7		70	S
8		0	S
9		100	
10		95	
11			
12			

The first approach is based on the COUNTA function. This simply returns the number of cells in a given range containing alphanumeric information. You might find it useful to recreate this table on a worksheet of your own.

Over to the right (i.e. NOT in columns A, B or C), find an empty cell, and enter the following formula:

```
=COUNTA(A:A)
```

This will return the number 5, because there are five cells in column A containing alphanumeric data. Because this data starts at Row 1, it ends at Row 5.

In the same way, you can use this on numerical information. In a different cell, enter:

```
=COUNTA(B:B)
```

This will return 10, representing the text header in column B plus the nine numerical values which are also in that column. Note that one of them is a zero, but this is still caught by COUNTA. Only a blank cell will not be caught. So the last data here is in Row 10.

But when we try to do the same with column C, we run into a problem. Try this formula:

```
=COUNTA(C:C)
```

This formula returns 7, although the data continues down to Row 8. The problem is, of course, that blank cell in C5. And here we find the limitation of the COUNTA method. If you suspect your list of values might contain blanks, you should avoid it, because it will tell you how many cells contain data, but not necessarily the location of the last one that does.

Let's instead look at a different approach, based on the MATCH function. MATCH is used normally to find the position of a value within a given range.

To try this out, in an empty cell on your sheet, type in:

```
=MATCH("Drew",A:A)
```

This returns the number 4, because the 4<sup>th</sup> cell in the range contains the value "Drew". Change the formula now, though, to this:

```
=MATCH("Charlie",A:A)
```

I'll bet you were expecting this formula to return 5, because Charlie is in the 5<sup>th</sup> cell in the range. But it actually returns 3. What's going on?

The answer is because the MATCH function has an optional third argument. In full, the format for this command is:

```
MATCH(lookup_value,lookup_array,match_type)
```

Of course, "lookup\_value" is what we want to find, and "lookup\_array" is the range in which we want to look. Both these arguments are needed in any MATCH formula. But what of the third argument? Here's what Office Help has to say:

**Match\_type** is the number -1, 0, or 1. Match\_type specifies how Microsoft Excel matches lookup\_value with values in lookup\_array.

- If match\_type is 1, MATCH finds the largest value that is less than or equal to lookup\_value. Lookup\_array must be placed in ascending order: ...-2, -1, 0, 1, 2, ..., A-Z, FALSE, TRUE.
- If match\_type is 0, MATCH finds the first value that is exactly equal to lookup\_value. Lookup\_array can be in any order.
- If match\_type is -1, MATCH finds the smallest value that is greater than or equal to lookup\_value. Lookup\_array must be placed in descending order: TRUE, FALSE, Z-A,...2, 1, 0, -1, -2,..., and so on.
- If match\_type is omitted, it is assumed to be 1.

Now this takes a minute to get your head around. Let's relate it to our problem, and you can see what happened. Because we omitted the third argument, MATCH worked on a Match\_type 1 basis, which meant it thought the values were in ascending order and looked for the "largest" value less than or equal to Charlie. In this context of course, think of "largest" as meaning the latest in alphabetical order. Let's step one value at a time through the range and follow the logic.

**Step 1:** the function arrives at A1. It knows that this is likely to be a column heading, so if it's not an exact match for "Charlie", it passes straight by.

**Step 2:** it gets to A2, which contains the string "Andy". This isn't a match, and it's not alphabetically later than the search string either, so the function moves on.

**Step 3:** now it comes to A3, which contains the string "Bill". Again, no match, and it comes alphabetically before the search string, so let's move on.

**Step 4:** it reaches A4, which contains the string "Drew". Now, this doesn't match the search string, but it's alphabetically later than the search string. Remember that with this Match\_type set, the MATCH formula assumes the range is sorted in alphabetical order, so it thinks there's no point in looking any further. It needs to go back one step to find the "largest" value that isn't greater than the search string.

So the formula reports "Bill" as the best match, which equates to a value of 3, because it's the third value in the range.

Don't type this in yet, but think about the value you think the formula below would return:

=MATCH(99,B1:B10)

Now type it in an empty cell, and see if you were right. The result you got should be equal to the total number of letters in the name of the author of this article.

How weird is that? The best match for "99" is zero? Only, of course, because the next value in the list is greater than 99, thus forcing the formula to stop and pick the immediately preceding value.

And here is the key to understanding how this use of MATCH to find the last cell with data actually works. If I tried to MATCH to an absolutely **enormous** number, the MATCH formula should always get to the last number in any range without a match. When this happens, the MATCH formula simply reports the position in the range of the last cell containing number data. So if there are 100 cells in the range, the formula will return the value 100, if there are 3,000, it will return the value 3,000, and so on, even if in the middle of the range there are blank cells. But, usefully for us, it will discount any blank cells at the end of the range. So if I specify an entire column, it will return the row number of the last cell in the column with a number in it.

Sounds good, but how big is "enormous". Well, the biggest positive number Excel can cope with is  $1 \times 10^{307} - 1$ . This is number with 307 digits, all of which are 9s. We can get pretty close to it with the following in Excel:

9.999999E306

That's the number "9999999..." followed by 300 zeros.

Let's put it all together. In an empty cell, type this:

```
=MATCH(9.999999E+306,B:B)
```

This time, the formula returns the value 10, because it is the 10<sup>th</sup> cell in the range which contains the last number value.

Now let's try it with a text column, though. Type this formula in:

```
=MATCH(9.999999E+306,A:A)
```

This returns an error, because the formula expects numbers and you've given it text to match against. Remember that MATCH always lets you get away with text in the first cell, because it assumes this is a heading, but beyond that cell, you need all numbers or blanks for this to work.

Fortunately, though, there is a different form of MATCH which will work for us with text. Type in this formula:

```
=MATCH("*",$A:$A,-1)
```

You'll notice it returns the value 5, the row of the last text entry in column A. You might want to think about how this works. Of course, the "\*" is a text wildcard, which means "any text string" and by setting the Match\_type as -1, we're asking MATCH to find the "lowest" value which meets the condition, assuming the list is in descending numerical order. Got it? Well, you're either there or you aren't. Either way, it works!

You might think that this form of the command could replace the numeric MATCH we tried first, with all those 9s. To see why it can't, type any number into cell A6. You see what happens? Your text MATCH formula still returns 5 rather than 6, because it's looking only for text strings, not numerical values.

On the other hand, leave the number there in A6 and try this formula again:

```
=MATCH(9.999999E+306,A:A)
```

A minute ago, this gave us an error, but now it's returning 6, the position of the last (and only) number in column A.

To put the final nail in this particular coffin, type the name "Edgar" into cell A7. Now it's the text form of MATCH that gives us the right answer (7), while the numeric MATCH insists the last cell is in row 6.

What we really need is a formula which will return the last non-blank cell in a range, irrespective of whether this contains text data or numeric data. I call it the 'Combination MATCH' formula. Here it is:

```
=IF(ISERROR(MATCH(9.999999E+306,A:A)),MATCH("*",A:A,-1),  
IF(ISERROR(MATCH("*",A:A,-1)),MATCH(9.999999E+306,A:A),  
MAX(MATCH(9.999999E+306,A:A),MATCH("*",A:A,-1))))
```

Let's deconstruct it. It says:

1. Try and get a result from the numeric MATCH formula first
2. If this produces an error, then there are no numbers in the range, so just use the text MATCH formula and go no further
3. Otherwise, try and use the text MATCH formula alone
4. If this produces an error, then there are no text strings in the range, so just use the numeric MATCH formula and go no further
5. If you get this far, it means there are numbers and text strings in the range, so calculate both versions of the MATCH formula and return whichever produces the highest value.

So there we are. By way of summary, let's look at what we've found for all the different formulae. By now, your data should look slightly different from the way it started:

	A	B	C
1	<b>Name</b>	<b>Name</b>	<b>Region</b>
2	Andy	90	S
3	Bill	80	S
4	Drew	60	N
5	Charlie	90	
6	1	50	N
7	Edgar	70	S
8		0	S
9		100	
10		95	
11			
12			

Applying the different methods to each column returns the following results:

Method	Column A	Column B	Column C
COUNTA	7	10	7
Numeric MATCH	6	10	#N/A
Text MATCH	7	1	8
Combination MATCH	7	10	8
Actual last row	7	10	8

As you will see, only the 'Combination MATCH' formula returns the actual last row in all situations.

So, we've built an approach to find the last cell containing data in a given range. We started by using the COUNTA formula, and found that this worked well provided that there were no blanks in the range. We moved on to the MATCH formula, and saw how in its numeric form and its text form it produced a reliable answer provided the range contained only the correct kind of data: either numbers or text strings. Finally, we built the Combination MATCH formula, to produce the right answer in all conditions.