

How to Count Cells with OR Criteria Using COUNTIF in Google Sheets: A Step-by-Step Guide

Authored by
stats writer

December 3, 2025

RECOMMENDED CITATION

stats writer (2025). *How to Count Cells with OR Criteria Using COUNTIF in Google Sheets: A Step-by-Step Guide*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=103875>

The standard COUNTIF function in Google Sheets is robust for counting cells based on a single condition. However, when spreadsheet tasks require counting cells that satisfy one condition **or** another--the classic Boolean OR operator logic--the standard syntax proves insufficient. To achieve this complex counting requirement, we must combine several powerful functions: COUNTIF, the ArrayFormula wrapper, and the SUM function. This combination creates a mechanism that processes multiple criteria simultaneously and aggregates the resulting counts into a single, accurate total, which is crucial for flexible data analysis.

The Challenge of Implementing OR Logic in COUNTIF

Analyzing datasets often requires identifying records that belong to one of several categories. For instance, you might need to count the number of sales representatives located in "East" or "West" regions, or tally transactions that meet price points of \$50, \$75, or \$100. While the COUNTIF function is excellent for single-criterion counting, it lacks a built-in parameter for handling multiple, alternative criteria directly within its argument structure. Attempting to input multiple criteria separated by commas, as one might instinctively try, often results in errors or misinterpretations of the range argument rather than successful OR evaluation.

This limitation necessitates a functional workaround that leverages the spreadsheet's ability to handle array processing. The key conceptual shift is moving from a single comparison operation to a collection of comparison operations executed simultaneously. Instead of asking Google Sheets, "Is this cell equal to A OR B?", we must structure the query as: "Calculate the count of cells equal to A, and separately calculate the count of cells equal to B, then add those two results together." This is precisely what the combined formula structure accomplishes, treating each criterion as an independent counting task before consolidation.

The technique presented here is the most standard and reliable method for achieving true OR operator functionality with counting functions in Google Sheets, ensuring that you can accurately summarize data based on complex logical requirements without manual filtering or concatenation of simple formulas. This structured approach is fundamental for anyone performing advanced data filtering and aggregation.

Understanding the Required Syntax for Multiple Criteria

To implement OR logic effectively using COUNTIF, we must utilize three components working in tandem: the ArrayFormula, the SUM function, and an array constant containing all the criteria. The ArrayFormula is the essential wrapper that allows the COUNTIF function to process the criteria array rather than just a single value. Without ArrayFormula, COUNTIF would only evaluate the first element of the criteria array, ignoring all subsequent conditions and resulting in an incorrect, partial count.

The general syntax is highly adaptable and requires the criteria to be enclosed within curly braces {} to signify an array. This array constant tells COUNTIF to perform its counting operation for each value listed inside the braces. For example, if you list three values, COUNTIF runs three separate counting operations on the specified range, generating three corresponding results--which is an array of individual counts.

Finally, the outer SUM function is responsible for taking the array of results generated by the ArrayFormula(COUNTIF(...)) combination and aggregating them into a single, meaningful total. Since the conditions are generally mutually exclusive (a single cell cannot simultaneously contain "Value1" and "Value2"), summing the counts accurately reflects the total number of cells that meet any one of the specified criteria. This structure is universally applicable for both text and numeric criteria.

The basic structure for counting cells in column A that meet criteria "Value1", "Value2", or "Value3" is defined as follows, demonstrating the necessary nesting of functions:

```
=ArrayFormula(SUM(COUNTIF(A:A,{"Value1", "Value2", "Value3"})))
```

This formula specifically counts the number of cells within the entire column A that are precisely equal to "Value1", "Value2", or "Value3." Notice that text criteria must be enclosed in double quotes, even within the array constant, adhering strictly to standard spreadsheet formula syntax conventions.

Example 1: Counting Cells Based on Text Criteria

We will now walk through a practical demonstration using regional data. Suppose we have a dataset listing sales figures and their corresponding geographic regions, and the goal is to quickly determine the total number of entries belonging to the "East" or "South" regions. This is a common requirement in regional performance summaries, where combining data from specific non-adjacent regions is necessary for aggregated reporting and identifying performance trends across specific markets.

Consider the following sample data established in Google Sheets. Column A contains the Region labels, and our analysis focuses solely on this range. The ability to use full column references (A:A) makes the formula highly dynamic and scalable, automatically adjusting if new data is appended to the sheet without requiring manual range updates.

	A	B	C	D	E
1	Division	Team	Points		
2	West	Mavericks	99		
3	South	Hawks	95		
4	West	Suns	98		
5	West	Spurs	105		
6	West	Rockets	103		
7	West	Lakers	114		
8	East	Celtics	109		
9	East	Hornets	85		
10	West	Blazers	91		
11	North	Knicks	95		
12	North	Bulls	99		
13	South	Magic	104		
14	South	Heat	103		
15					
16					
17					
18					
19					

To count the number of cells in column A that contain either "East" or "South," we construct the criteria array using these two specific text strings. The formula integrates the range (A:A) and the array constant {"East", "South"} within the COUNTIF function, ensuring that both conditions are checked sequentially by the ArrayFormula processing engine.

The resulting formula that achieves this OR counting logic is:

=ArrayFormula(SUM(COUNTIF(A:A,{"East", "South"})))

When this formula is executed, COUNTIF first counts entries matching "East" (3 entries) and then counts entries matching "South" (2 entries). This generates the intermediate array {3, 2}. The outer SUM function then adds these elements together, yielding the final result 5. The screenshot below confirms the successful application of this logic in the spreadsheet interface, showing the total count in cell D1.

	A	B	C	D	E
1	Division	Team	Points		
2	West	Mavericks	99		5
3	South	Hawks	95		
4	West	Suns	98		
5	West	Spurs	105		
6	West	Rockets	103		
7	West	Lakers	114		
8	East	Celtics	109		
9	East	Hornets	85		
10	West	Blazers	91		
11	North	Knicks	95		
12	North	Bulls	99		
13	South	Magic	104		
14	South	Heat	103		
15					
16					
17					
18					
19					

As demonstrated, a total of **5** cells have a value of "East" or "South" in column A, confirming the effectiveness and accuracy of combining the array processing functions for complex OR criteria.

Example 2: Counting Cells Based on Numeric Criteria

The OR logic technique is not limited to text; it is equally effective when counting based on specific numerical values. Suppose column C in our dataset represents specific item scores or revenue amounts, and we are interested in counting how many entries fall precisely into the high-value thresholds of 95, 99, or 103. This is essential for auditing or performance analysis where only specific benchmark numbers are relevant for success metrics.

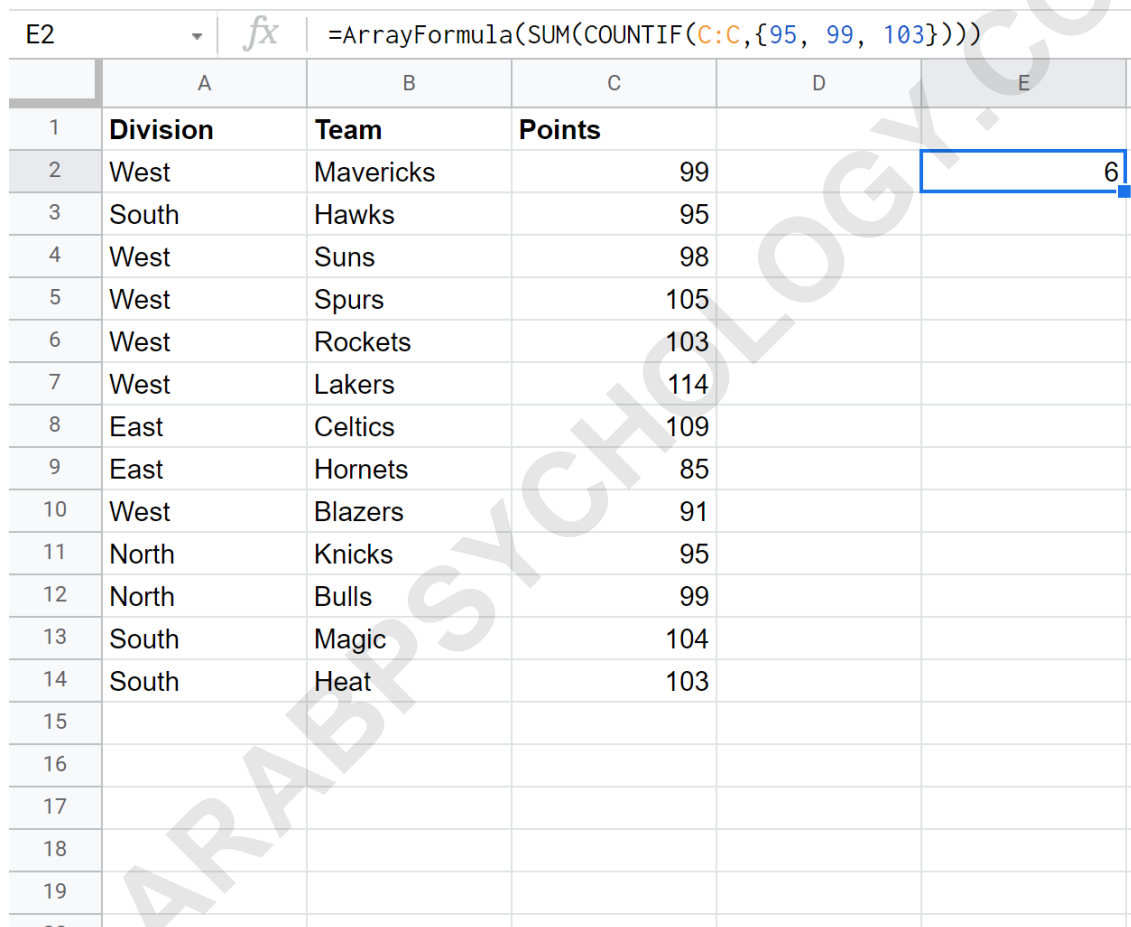
When dealing with numerical criteria within the array constant, we typically omit the quotation marks, as the values are meant to be treated as numbers for comparison. This small distinction simplifies the formula input and adheres to standard numerical comparison practices within spreadsheet environments, ensuring that the counts are based on actual numeric equality.

To count the number of cells in column C that have a value of 95, 99, or 103, the formula structure remains identical to the text example. Only the target range and the content of the criteria array change:

=ArrayFormula(SUM(COUNTIF(C:C,{95, 99, 103})))

This execution yields the count for each number independently (e.g., how many cells equal 95, how many equal 99, and how many equal 103) before the SUM function provides the final aggregated result. This robust handling of numeric criteria demonstrates the flexibility of the ArrayFormula approach in various data analysis scenarios, regardless of data type.

The following screenshot shows the formula applied to the dataset, successfully counting the occurrences of the specified numeric values in column C:



The screenshot shows a Google Sheet with a table of basketball teams and their points. The formula in cell E2 is `=ArrayFormula(SUM(COUNTIF(C:C,{95, 99, 103})))`. The result in cell E2 is 6. The table data is as follows:

	A	B	C	D	E
1	Division	Team	Points		
2	West	Mavericks	99		6
3	South	Hawks	95		
4	West	Suns	98		
5	West	Spurs	105		
6	West	Rockets	103		
7	West	Lakers	114		
8	East	Celtics	109		
9	East	Hornets	85		
10	West	Blazers	91		
11	North	Knicks	95		
12	North	Bulls	99		
13	South	Magic	104		
14	South	Heat	103		
15					
16					
17					
18					
19					
20					

Advanced Considerations: The Distinction Between OR and AND Logic

It is crucial to differentiate between implementing OR operator logic and AND logic in Google Sheets. The `COUNTIF + ArrayFormula` method demonstrated above is strictly for OR conditions (counting if Criterion A is met OR Criterion B is met). This assumes that the criteria apply to the **same range** and that we are interested in counting entries that match *any* of the conditions.

In contrast, achieving AND logic (counting if Criterion A is met AND Criterion B is met) typically

requires the use of the [COUNTIFS](#) function. The [COUNTIFS](#) function inherently supports multiple criteria across multiple ranges (or the same range) but treats them conjunctively (AND). It is vital not to confuse these two logical requirements, as the syntax and resulting counts will be fundamentally different. For instance, using `COUNTIFS(A:A, "East", C:C, 95)` counts rows where Column A is "East" *and* Column C is 95, a very different outcome than our OR approach.

While the [COUNTIF](#) + [ArrayFormula](#) approach is powerful for OR logic, it is specifically designed for equality comparisons (finding exact matches). For more complex comparisons involving greater than (>), less than (<), or wildcards, the criteria within the array constant must be constructed carefully, potentially requiring concatenation if they involve cell references or variables to ensure the correct string is passed to COUNTIF.

Alternative Method: Utilizing the QUERY Function

For users who prefer SQL-like syntax or need extremely flexible criteria (including comparisons like greater than, pattern matching, or combining AND/OR logic), the [QUERY](#) function offers a powerful alternative. The QUERY function allows the use of a standard SQL `WHERE` clause, which natively supports complex Boolean logic using both `AND` and `OR` operators in a highly readable format.

To replicate the OR counting from Example 1 using QUERY, where we counted "East" or "South" entries in column A, the syntax would be dramatically different but perhaps more intuitive for those familiar with databases. The general structure involves selecting the count where the target column matches the OR criteria:

```
=QUERY(A:A, "SELECT COUNT(A) WHERE A = 'East' OR A = 'South' LABEL COUNT(A) ")
```

The primary advantage of the [QUERY](#) function is its enhanced readability and ease of maintenance when the number of OR criteria becomes very large or when combining OR and AND conditions within the same filter. While the [COUNTIF](#) array approach is efficient and direct for simple OR counting, QUERY provides a more structured and transparent way to handle multiple nested conditions, making it a highly viable alternative for advanced data analysis tasks.

Summary of Key Implementation Steps

Mastering the combination of [COUNTIF](#), [ArrayFormula](#), and [SUM](#) is essential for implementing OR logic in Google Sheets. Following these best practices ensures formula accuracy and reliability, turning a simple function into a sophisticated counting tool:

Use Curly Braces: Always enclose your multiple criteria within curly braces `{ }` to create the necessary [array constant](#). This defines the set of conditions to be evaluated.

Ensure Text Quotation: Remember to wrap text criteria (e.g., "East") in double quotes, even

when inside the curly braces. Numerical criteria generally do not require quotes unless they are stored as text.

Apply ArrayFormula: The entire structure must be wrapped in `ArrayFormula()` to enable iterative processing of the criteria array by the COUNTIF function. This is the engine that generates the array of results.

Utilize SUM: The outer `SUM` function is mandatory to collapse the array of individual counts into a single, final total, providing the final aggregated count based on the OR condition.

By applying these techniques, spreadsheet users can move beyond simple single-criterion analysis, enabling detailed reporting and robust conditional summaries required for effective data management and decision-making.

Further Resources for Conditional Counting

For expanding your skills in conditional data manipulation within Google Sheets, consider exploring related functions and tutorials that cover other common counting and summing operations. These functions often serve as alternatives or complements to the `COUNTIF` OR array method:

How to use the `COUNTIFS` function for handling AND logic across multiple columns simultaneously.

Techniques for using `SUMIF` and `SUMIFS` to aggregate numerical values based on single or multiple criteria, rather than just counting occurrences.

Methods for counting unique values or cells containing partial text matches using wildcards in conjunction with array formulas for fuzzy matching.

The following tutorials explain how to perform other common `COUNTIF()` operations in Google Sheets: