Learning to Count with Wildcards: A Guide to COUNTIF in Google Sheets

Authored by

Mohammed looti

October 30, 2025

Harnessing the full potential of Google Sheets demands a mastery of its functional library. Among the most crucial and versatile functions are COUNTIF and COUNTIFS, particularly when they are dynamically combined with wildcard characters. This powerful combination allows spreadsheet users to move beyond the limitations of exact text matches, enabling sophisticated data analysis by counting cells based on partial text matches or specific, flexible patterns.

This comprehensive guide is designed to walk you through the precise methodology of integrating wildcards into your counting formulas within Google Sheets. We will thoroughly explore the fundamental concepts behind these special characters, provide detailed, practical examples for both COUNTIF and COUNTIFS, and offer advanced insights into techniques that will help you efficiently manage, filter, and interpret large datasets with greater precision.

Understanding Wildcard Characters in Google Sheets

Before implementing these powerful tools in specific formulas, it is essential to establish a clear understanding of what wildcard characters are and how they operate within the specialized environment of Google Sheets. Wildcards are special symbols that function as placeholders, representing either a single unknown character or an entire sequence of unknown characters. This capability makes them indispensable for pattern matching within your search criteria, allowing you to define highly flexible search parameters that are not restricted to rigid, character-for-character text comparisons.

The flexibility offered by wildcards significantly enhances the querying capabilities of spreadsheet applications. They allow users to construct dynamic searches, saving substantial time and effort when navigating large volumes of data or dealing with inconsistent data entry formats. By understanding the specific roles of the primary wildcards, you gain the ability to structure search criteria that are either broadly inclusive or highly precise, depending on your analytical needs.

Google Sheets primarily recognizes and supports two distinct types of wildcard characters for use within functions like COUNTIF and COUNTIFS:

Asterisk (*): This wildcard is used to represent any sequence of zero or more characters. It provides the broadest matching capability. For instance, if you define the pattern as `"apple*"`, it will successfully match "apple", "apples", "applepie", and any other string starting with "apple". Conversely, `"*apple"` would match terms like "redapple" or "greenapple". When deployed as `"*apple*"`, it matches any cell that contains the substring "apple" anywhere within its text, making it ideal for general substring searches.

Question Mark (?): This wildcard specifically represents any single character. The question mark is particularly useful when you know the exact length of the desired string but are unsure about one or more specific characters within it. For example, the criterion `"b?t"` would accurately match "bat", "bet", "bit", or "but", but it would exclude longer words such as "boat" or "boot" because the

pattern only permits one character in the middle position.

The COUNTIF Function: Review and Wildcard Integration

The COUNTIF function serves the fundamental purpose of counting the number of cells within a specified range that satisfy a single, defined criterion. Its straightforward syntax is `COUNTIF(range, criterion)`. While it is frequently utilized for simple exact matches--such as counting cells that are precisely equal to "Complete"--its true potential for dynamic filtering is unlocked when it is combined with wildcard characters.

By integrating wildcards into the criterion argument, the COUNTIF function is fundamentally transformed from a rigid comparison tool into a flexible and powerful pattern matching utility. This functionality enables complex searches, such as counting all product descriptions that contain a specific model number, or aggregating survey responses that share a common keyword. We will now proceed to explore practical applications using a single wildcard pattern to count cells based on partial text matches.

Example 1: COUNTIF with One Wildcard Criterion

A common scenario in data analysis requires the ability to count all cells within a designated range that contain a specific substring, irrespective of any characters that may precede or follow it. This technique is invaluable for tasks such as filtering inventory records, tracking project names, or identifying all transactions related to a specific vendor code embedded within longer descriptive text.

To achieve this generalized substring search, we utilize the * wildcard strategically placed both before and after our target substring. The following formula provides a clear illustration of how to count cells that contain the substring "string" anywhere within the data located in the specified range:

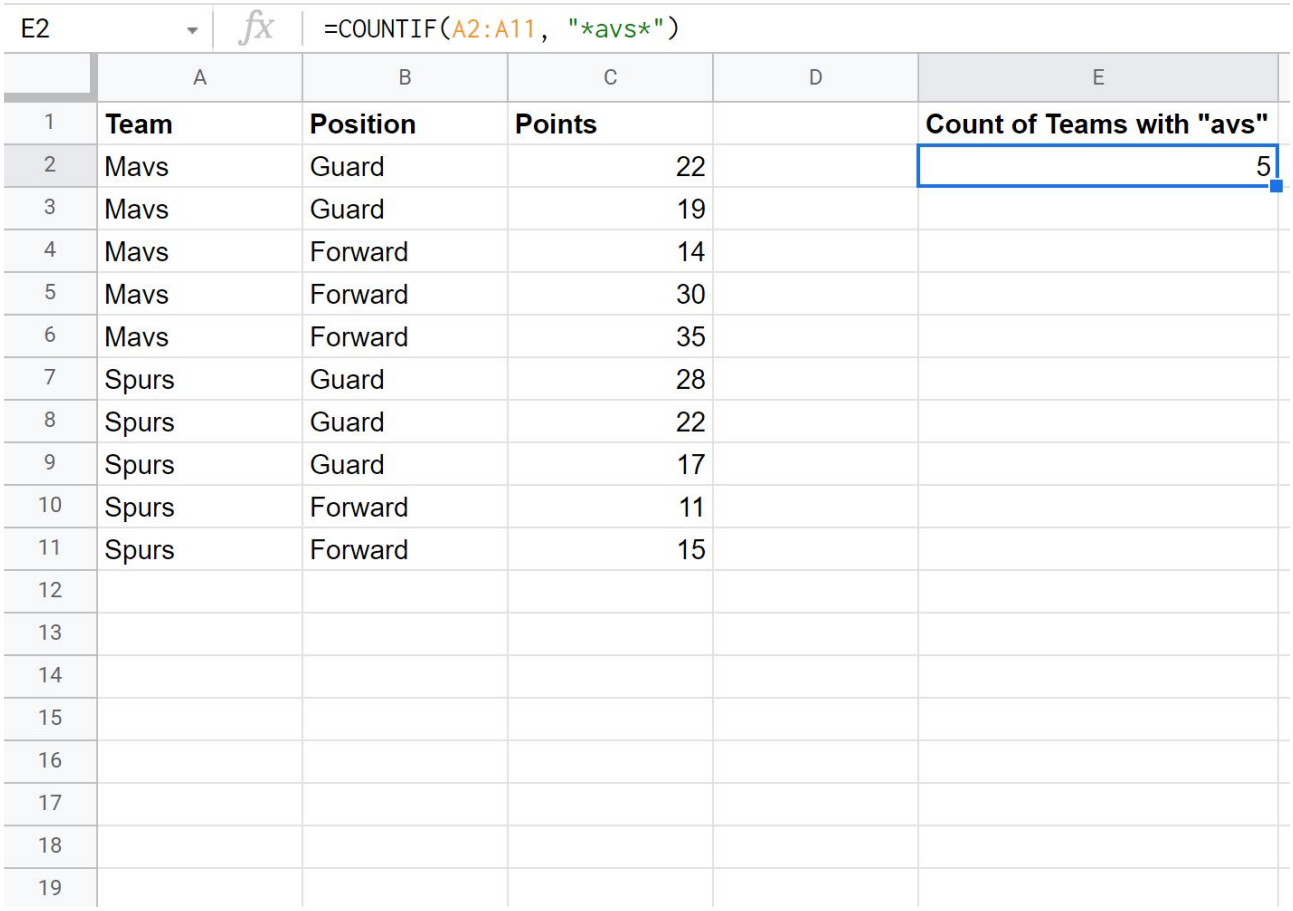
=COUNTIF(A2:A11, "*string*")

This formula precisely instructs Google Sheets to meticulously evaluate every cell across the column segment **A2:A11**. The criterion `"*string*"` functions as a flexible search pattern: the initial * matches any preceding characters (or none), and the subsequent * matches any succeeding characters (or none). The result is an accurate count of all cells where "string" exists as an embedded substring.

To demonstrate this in a live example, suppose our task is to determine the count of cells in column A that contain the sequence of letters "avs" at any position within the cell's content. We would confidently apply the following formula, demonstrating efficiency over manual filtering:

=COUNTIF(A2:A11, "*avs*")

The subsequent screenshot visually confirms the application of this formula and illustrates the resulting output, showcasing the effectiveness of the partial match:



	A	B	C	D	E
1	**Team**	**Position**	**Points**		**Count of Teams with "avs"**
2	Mavs	Guard	22		5
3	Mavs	Guard	19		
4	Mavs	Forward	14		
5	Mavs	Forward	30		
6	Mavs	Forward	35		
7	Spurs	Guard	28		
8	Spurs	Guard	22		
9	Spurs	Guard	17		
10	Spurs	Forward	11		
11	Spurs	Forward	15		
12					
13					
14					
15					
16					
17					
18					
19					

As clearly depicted, the formula successfully identified and tallied **5** cells within the targeted range **A2:A11** that incorporate the substring "avs". This perfectly illustrates the efficiency gained by using a single wildcard criterion to execute complex partial text searches.

Example 2: COUNTIFS with Multiple Wildcard Criteria

While COUNTIF is excellent for analyzing data based on a single condition, practical data analysis frequently necessitates evaluating multiple, simultaneous criteria. This is the domain where the COUNTIFS function proves indispensable. COUNTIFS is specifically engineered to count cells only when they satisfy all specified conditions across various ranges. Its syntax requires pairs of arguments: `COUNTIFS(range1, criterion1, , ...)`, where each pair imposes an additional requirement that must be met.

The true advantage of COUNTIFS, especially when utilized with wildcard characters, is its robust capability to filter data based on multiple partial matches across different columns. For example, a user might need to count all inventory records where the Supplier column contains "East" AND the Status column contains "Pending", regardless of any other surrounding text in those cells.

Let us examine a concrete example where we aim to count rows where column **A2:A11** contains the substring "avs" and, concurrently, column **B2:B11** contains the substring "Gua". The required formula to perform this two-dimensional partial match filtering is:

=COUNTIFS(A2:A11, "*avs*", B2:B11, "*Gua*")

Within this syntax, the first set of arguments (**A2:A11**, `"*avs*"`) verifies the presence of "avs" in the first data column, while the second set (**B2:B11**, `"*Gua*"`) confirms the presence of "Gua" in the second column. The key operational principle here is the logical AND inherent in COUNTIFS: only those rows where BOTH conditions evaluate to true will be aggregated in the final count. This provides a mechanism for highly targeted, multi-criteria data segmentation.

Please refer to the following screenshot to see this complex filtering operation executed and observe its precise numerical outcome:

E2 fx =COUNTIFS(A2:A11, "*avs*", B2:B11, "*Gua*")

	A	B	C	D	E	F
1	**Team**	**Position**	**Points**		**Team with "avs" and Position with "gua"**	
2	Mavs	Guard	22		2	
3	Mavs	Guard	19			
4	Mavs	Forward	14			
5	Mavs	Forward	30			
6	Mavs	Forward	35			
7	Spurs	Guard	28			
8	Spurs	Guard	22			
9	Spurs	Guard	17			
10	Spurs	Forward	11			
11	Spurs	Forward	15			
12						
13						
14						
15						
16						
17						
18						
19						

The result clearly confirms that there are exactly **2** rows within the entire dataset where the "Team"

column (A) contains "avs" and the "Position" column (B) simultaneously contains "Gua". This clearly underscores how the synergy between COUNTIFS and wildcards delivers a robust solution for multi-criteria pattern matching and granular data analysis.

Advanced Wildcard Techniques and Considerations

To master the use of wildcards, it is necessary to go beyond the basic `*` and `?` functionalities and understand several advanced techniques and critical operational considerations. These nuances are vital for ensuring absolute accuracy and leveraging the full potential of counting functions in complex data environments.

Escaping Wildcards: A crucial technique involves scenarios where your search criterion contains a literal * or ? that you need to be treated as a character itself, rather than a functional wildcard. Google Sheets allows you to "escape" these symbols by prefixing them with a tilde (~). For example, if you must count cells that literally contain the string "Q&A?", your criterion should be written as `"Q&A~?"`. Similarly, to search for "Product*ID" literally, you would use `"Product~*ID"`.
Combining Wildcards: For highly specified searches, you have the ability to combine both the asterisk and ? wildcards within a single criterion to create intricate patterns. For instance, the pattern `"A?B*C"` would successfully match strings like "AxByzC", "A1B_dataC", or "ADBCC". However, it would intentionally exclude "AB_dataC" because the question mark requires the presence of exactly one character between 'A' and 'B'. This high degree of flexibility permits the creation of customized searches based on the structural requirements of your data entries.
Case Sensitivity: It is imperative to remember that, by default, COUNTIF and COUNTIFS functions in Google Sheets are generally case-insensitive when evaluating text criteria, even when those criteria include wildcards. This means that a search for `"*apple*"` will match "Apple", "apple", or "APPLE" interchangeably. If your analytical requirements mandate strictly case-sensitive counting with wildcards, the methodology typically requires integrating more advanced functions, such as the REGEXMATCH function, nested within other array formulas.
Performance Considerations: While the use of wildcards is exceptionally powerful, applying overly complex wildcard criteria across extremely large ranges (e.g., hundreds of thousands of rows) may potentially degrade spreadsheet performance. For the vast majority of standard datasets, this is not a concern, but efficiency should be monitored when working with massive data volumes and numerous formula calculations.

Mastering these advanced concepts allows you to conduct more robust and precise data queries, significantly enhancing the efficiency and effectiveness of your Google Sheets workflows.

Common Use Cases and Best Practices

The strategic combination of COUNTIF/COUNTIFS with wildcard characters opens up a vast array

of practical applications across diverse sectors, including market research, inventory control, and general data management tasks. These functions provide critical tools for quickly summarizing and segmenting data based on partial information.

Filtering Survey Responses: When analyzing qualitative data, respondents often provide varied phrasing (e.g., "Yes, definitely," or "I agree, yes"). Using a criterion like `"*Yes*"` ensures that you accurately count all affirmative responses, regardless of the surrounding descriptive text. This is a crucial step for efficient qualitative data analysis.

Inventory Management and Product Categorization: For product databases where IDs or descriptions follow a structured but varied naming convention (e.g., "MTR-101-RED", "MTR-205-BLUE"), you can quickly tally all items belonging to the "MTR" series using `"MTR*"`. This partial matching is highly effective for rapid categorization and reporting.

Categorizing Text Data: In scenarios involving large volumes of unstructured text, such as customer feedback logs or document titles, wildcards simplify the categorization process. You can count all feedback entries containing "delivery issues" or "billing error" even if the exact wording differs slightly.

Identifying Data Anomalies: While more sophisticated validation often requires REGEXMATCH, wildcards can be used for initial checks, such as using `"?????"` to count entries that have a specific, required length (using the ? wildcard).

When deploying these functions, it is a best practice to meticulously verify your defined ranges and criteria. Ensure that your wildcard placement is accurate to capture the intended patterns without accidentally including or excluding relevant data. Conducting regular test runs on a small, representative subset of your data will help validate the formula logic before applying it universally across your entire dataset.

Conclusion

The capacity to integrate wildcard characters within the COUNTIF and COUNTIFS functions fundamentally elevates the way you interact with and derive insights from data in Google Sheets. These functions empower users to transcend the limitations of simple exact matches, providing flexible and robust pattern matching capabilities that are indispensable for effective modern data management.

By thoroughly understanding the distinct roles of the * and ? wildcards, and knowing how to apply them for both single-condition (COUNTIF) and multiple-condition (COUNTIFS) analysis, you gain a significant competitive advantage in handling diverse and sometimes inconsistent datasets. Remember to integrate advanced techniques, such as escaping wildcards for literal searches and maintaining awareness of case sensitivity. Incorporate these powerful tools into your daily workflow to unlock unparalleled levels of efficiency and precision in all your spreadsheet tasks.

Additional Resources

For users committed to deepening their expertise in Google Sheets and exploring further advanced data manipulation techniques, the following related topics offer valuable insights into other common and powerful operations, including the use of advanced regular expressions via REGEXMATCH for even more complex pattern matching: