

How to Easily Sum Filtered Rows in Excel Using SUBTOTAL

Authored by
stats writer

November 30, 2025

RECOMMENDED CITATION

stats writer (2025). *How to Easily Sum Filtered Rows in Excel Using SUBTOTAL*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=102764>

The necessity of summing data based on criteria is a common task when working with large data sets in Excel. While the standard **SUM** function aggregates all values in a range, it fundamentally fails when the data is restricted using an applied filter. To accurately perform calculations like summation, averaging, or counting on visible rows only, users must employ the specialized SUBTOTAL function. This powerful tool is essential for effective data analysis when working with dynamic or filtered ranges.

The procedure requires two main steps: first, applying a filter to the target range to define the visible subset of data, and second, inserting the **SUBTOTAL** formula into the desired cell. For instance, if the objective is to sum numerical values in column B, the correct formula structure would be `=SUBTOTAL(109, B:B)`. The inclusion of the argument 109 specifically instructs Excel to calculate the sum, ensuring that only the rows currently visible--those meeting the filtering criteria--are included in the final aggregation.

The Core Mechanism: Understanding SUBTOTAL Syntax

The core principle behind calculating the total of a filtered range relies entirely on the **SUBTOTAL** function. Unlike the basic **SUM()** function, which ignores whether rows are hidden by a filter, **SUBTOTAL()** is designed to adapt its calculation based on the current display state of the worksheet. This makes it the definitive choice for generating dynamic summaries of filtered data.

To implement this successfully, you must specify two crucial function arguments: the function number (which dictates the type of calculation) and the reference range (the data you wish to process). The standard syntax for summing a filtered range is as follows:

SUBTOTAL(109, A1:A10)

Crucially, the value **109** serves as the specific function code (or `function_num`) required for calculating the **sum** of a filtered range of rows. By utilizing the '100 series' codes (101-111), Excel explicitly excludes values that are hidden manually or hidden by filtering, ensuring accuracy in your visible data summaries.

The following detailed sections will demonstrate the practical application of this powerful function through a step-by-step example using sports data.

Comprehensive Guide to SUBTOTAL Function Codes

The utility of the **SUBTOTAL** function extends far beyond simple summation. It is a versatile tool capable of performing eleven different types of aggregation, each assigned a specific numerical code. These codes are categorized into two groups: the standard series (1-11) and the dynamic

'hidden row exclusion' series (101-111).

When you use codes 1 through 11, the function calculates results for visible cells but ignores rows hidden by filtering. However, if rows are manually hidden (by selecting rows, right-clicking, and choosing 'Hide'), codes 1-11 will still include them in the calculation. This is where the 100 series becomes essential.

To guarantee that both filtered rows and manually hidden rows are definitively excluded from the calculation, analysts must utilize the 100 series of function numbers. For summing visible data, **109 (SUM)** is the code to use. Other frequently used codes in the 100 series include:

101: AVERAGE (Calculates the average of the visible values).

102: COUNT (Counts the number of cells containing numerical values in the visible range).

103: COUNTA (Counts the number of non-empty cells in the visible range).

104: MAX (Returns the largest value in the visible range).

105: MIN (Returns the smallest value in the visible range).

Choosing the correct function number based on the desired output is the most critical step in implementing the **SUBTOTAL** formula effectively. Always opt for the 100 series (e.g., 109 instead of 9) when dealing with data that may contain rows hidden through filtering or manual selection, thus ensuring maximum control over your summarized data.

Example Setup: Preparing the Data Set

To illustrate the efficiency of the **SUBTOTAL** function, let us utilize a practical scenario involving sports statistics. Suppose we are analyzing a detailed data set concerning various professional basketball teams. This data includes player information, their associated team, and their accumulated points, structured neatly across columns.

Our goal is to dynamically calculate the total points scored for only specific teams after applying a filter. This requirement necessitates moving beyond simple static formulas and leveraging Excel's built-in data aggregation capabilities to handle the visibility changes.

	A	B	C	D	E
1	Team	Points			
2	Mavs	99			
3	Mavs	94			
4	Warriors	93			
5	Celtics	97			
6	Mavs	104			
7	Warriors	109			
8	Celtics	99			
9	Warriors	84			
10	Celtics	89			
11					
12					
13					
14					
15					
16					
17					
18					
19					

Applying the Data Filter Framework

Our next step is to restrict the visible data to focus exclusively on players associated with certain teams. This filtering process is essential to create the environment where the **SUBTOTAL** function proves its worth by ignoring the excluded rows.

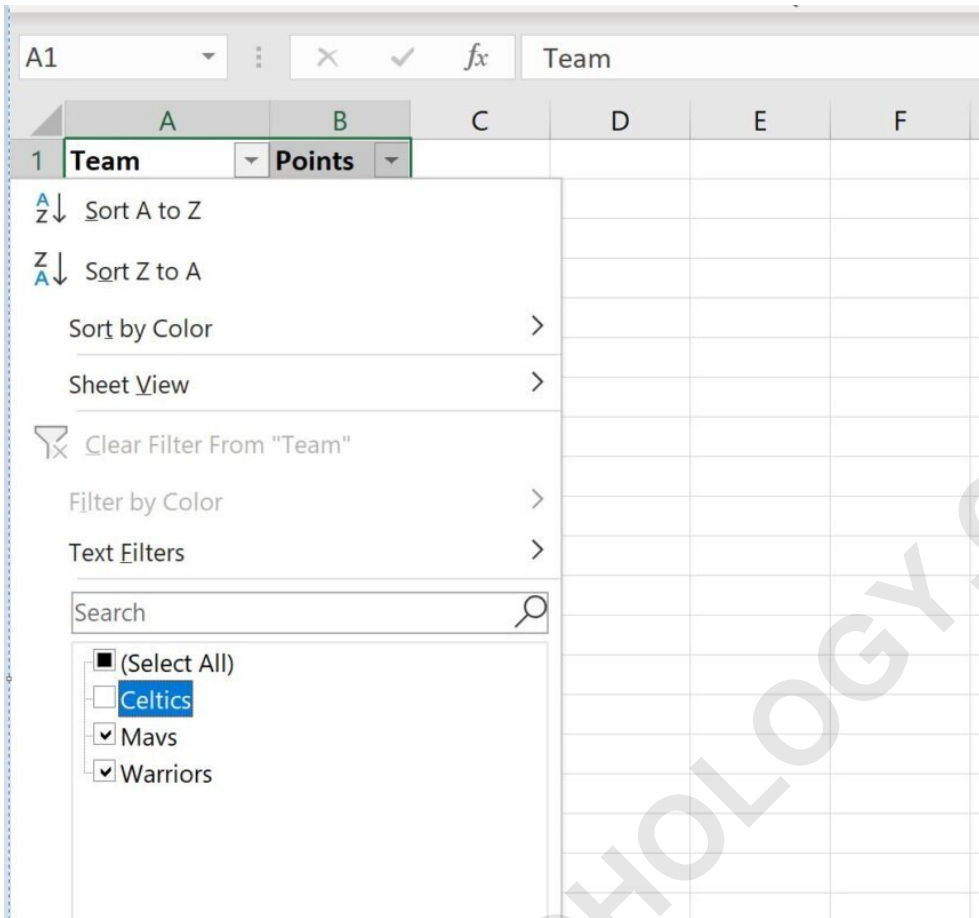
To begin the filtering process, first select the entire cell range that contains the data, for example, the range **A1:B10**. Then, navigate to the **Data** tab located in the top ribbon menu. Within the Data tools section, click the **Filter** button. This action will place dropdown arrows next to each column header, enabling interactive data selection.

	A	B	C	D	E	F	G	H
1	Team	Points						
2	Mavs	99						
3	Mavs	94						
4	Warriors	93						
5	Celtics	97						
6	Mavs	104						
7	Warriors	109						
8	Celtics	99						
9	Warriors	84						
10	Celtics	89						
11								
12								
13								
14								
15								

Executing the Filtering Criteria

With the filtering mechanism active, we now define our selection criteria. Let's filter the data to only show the players on the Mavs or the Warriors. This means we must explicitly deselect any other teams present in the column, such as the Celtics.

Click the dropdown arrow next to the **Team** column header. In the list of team names, locate and uncheck the box corresponding to "Celtics." This action specifies that all rows containing "Celtics" in the Team column should be hidden from view. After modifying the selection, click the **OK** button to apply the changes and update the visible data set.



Observing the Filtered Results

The data will automatically be filtered to remove rows where the team is "Celtics." The resulting table now shows only the subset of data meeting our criteria (Mavs and Warriors players). Notice that the row numbers are no longer sequential, indicating that certain rows have been effectively concealed by the filter.

This refined data view provides the perfect environment to test the critical difference between standard summation and the specialized **SUBTOTAL()** function. Only the relevant data points remain visible for accurate aggregation.

	A	B	C	D	E
1	Team	Points			
2	Mavs	99			
3	Mavs	94			
4	Warriors	93			
6	Mavs	104			
7	Warriors	109			
9	Warriors	84			
11					
12					
13					
14					
15					
16					
17					
18					
19					

Analyzing the Failure of the Standard SUM Function

It is essential to understand why a simple **SUM()** calculation is insufficient when summarizing filtered data. If we attempt to use the **SUM()** function (e.g., `=SUM(C2:C10)`) to aggregate the points column of the filtered rows, it will unfortunately return the sum of all of the original values, including those hidden by the filter.

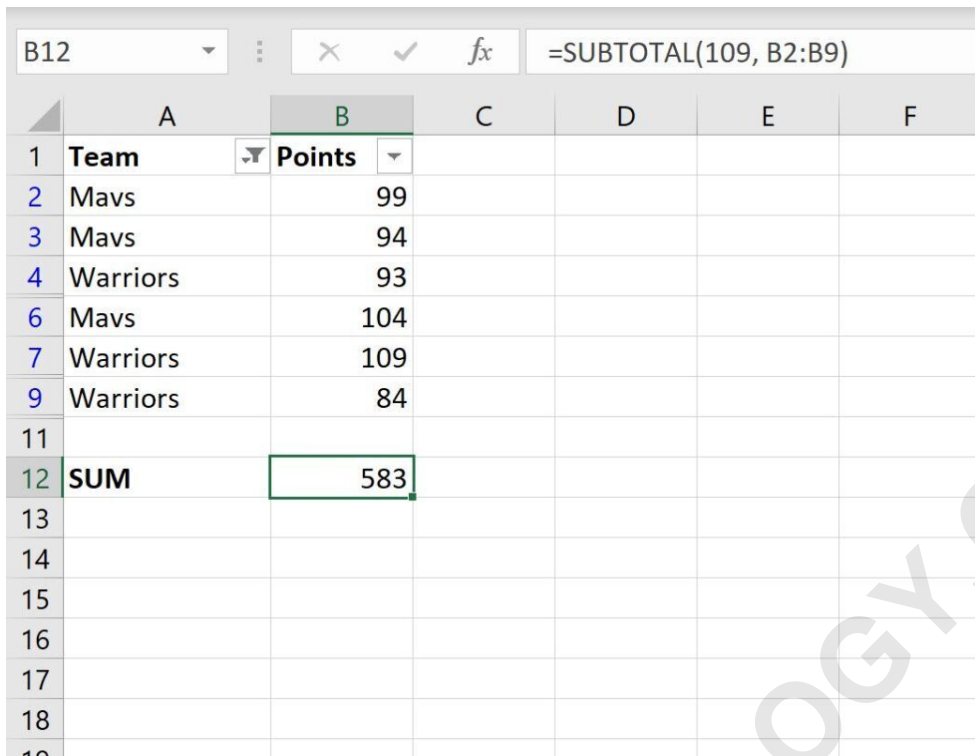
The standard **SUM()** function operates on the static cell address range, completely disregarding the visibility status of the rows set by the filter. This results in an aggregated total that is misleading when the intention is to summarize only the currently displayed data, as illustrated below where the total incorrectly includes the points from the hidden Celtics players:

	A	B	C	D	E
1	Team	Points			
2	Mavs	99			
3	Mavs	94			
4	Warriors	93			
6	Mavs	104			
7	Warriors	109			
9	Warriors	84			
11					
12	SUM	779			
13					
14					
15					
16					
17					
18					
19					
20					
21					

Achieving Accuracy with SUBTOTAL(109)

Instead of relying on the static **SUM()** function, we must implement the dynamic **SUBTOTAL()** function with the appropriate function code. By entering the formula `=SUBTOTAL(109, C2:C10)`, we instruct Excel to specifically perform a summation that recognizes and excludes hidden rows resulting from the application of the filter.

The use of the 100 series code ensures that the calculation adapts dynamically. The result immediately updates to reflect only the total points scored by the Mavs and Warriors players who are currently visible in the data set.



	A	B	C	D	E	F
1	Team	Points				
2	Mavs	99				
3	Mavs	94				
4	Warriors	93				
6	Mavs	104				
7	Warriors	109				
9	Warriors	84				
11						
12	SUM	583				
13						
14						
15						
16						
17						
18						
19						

Verification of Results

The value returned by the **SUBTOTAL(109, range)** formula represents the true aggregated total of the visible rows. This confirms the successful use of the function to perform calculations dynamically based on the current filtering state.

We can manually verify this outcome by taking the sum of the visible scores, which confirms the accuracy achieved by using the function code 109:

Sum of Points in Visible Rows: $99 + 94 + 93 + 104 + 109 + 84 = 583$.

Summary and Best Practices

The **SUBTOTAL** function is an indispensable tool for advanced data manipulation in Excel. Its capability to operate only on visible cells ensures that your summaries remain accurate and relevant, regardless of how aggressively the underlying data is filtered. Always remember to use the function numbers in the 100 series (101-111) when your objective is to exclude both manually hidden rows and rows hidden by filtering.

For routine financial reporting, statistical analysis, or large-scale data cleansing operations, mastering **SUBTOTAL(109, range)** will save significant time and prevent calculation errors associated with static summing functions. Integrate this technique into your workflow to ensure

maximum efficiency when handling complex datasets.

ARABPSYCHOLOGY.COM