

Learning to Use SUMIF: How to Sum Values Less Than a Specific Number in Excel

Authored by
Mohammed looti

June 3, 2026

RECOMMENDED CITATION

Mohammed looti (2026). *Learning to Use SUMIF: How to Sum Values Less Than a Specific Number in Excel*. PSYCHOLOGICAL STATISTICS. Retrieved from <https://statistics.arabpsychology.com/?p=3688>

The ability to perform conditional calculations represents a fundamental requirement for effective data analysis and reporting within [Microsoft Excel](#). At the heart of this functionality lies the powerful [SUMIF function](#), a tool designed to selectively aggregate numerical values. Specifically, **SUMIF** enables users to sum values located within a designated [range](#) that precisely match a specified [criteria](#). This comprehensive guide will focus intently on one of its most practical applications: how to use **SUMIF** to calculate the total of all values that are strictly **less than** a predefined numerical threshold.

Acquiring proficiency in this specific technique dramatically enhances your capacity for large-scale data processing. It allows you to quickly extract actionable, meaningful insights from extensive [datasets](#). Whether your responsibilities involve monitoring sales performance, meticulously tracking inventory levels, or ensuring strict adherence to budgetary limits, the skill of conditionally summing data based on a "less than" requirement is an indispensable asset for any serious Excel user.

Understanding the SUMIF Function for Conditional Summation

The [SUMIF function](#) is explicitly engineered to perform summation based on a singular, user-defined condition. To utilize this function effectively, it is essential to understand its fundamental structure. The standard syntax for **SUMIF** is presented below:

```
=SUMIF(range, criteria, )
```

Each argument plays a critical role in defining the scope and nature of the calculation:

range: This mandatory argument defines the group of cells that Excel must evaluate against your specified [criteria](#). It can encompass numerical values, textual data, or named ranges within your spreadsheet.

criteria: This is the core condition that determines which cells within the **range** will be included in the sum. The [criteria](#) can be expressed as a fixed number, an algebraic expression, a [cell reference](#), text, or even another function result. To implement a "less than" condition, you must employ the appropriate [comparison operator](#) combined with a value (e.g., "<100").

(optional): This argument specifies the actual cells to be summed. If this argument is intentionally omitted, the function defaults to summing the cells contained within the initial [range](#) argument.

Crucially, whenever you apply a "less than" condition using the [SUMIF function](#), your [criteria](#) must begin with the less than symbol (<). Because this symbol is recognized as a [comparison operator](#) within Excel's text parsing engine, it must be encapsulated within double quotation marks (e.g., "<") when combined with a numerical value or a [cell reference](#).

Basic Application: Summing Values Less Than a Fixed Number

We begin with the most straightforward implementation of the [SUMIF function](#). This scenario involves summing all numerical entries within a defined [range](#) that fall below a specific, hardcoded number. The foundational [formula](#) required for this precise operation is structured as follows:

=SUMIF(B2:B13, "<100")

In this specific example, the [formula](#) issues a clear directive to Excel: calculate the aggregate sum of all numerical values found exclusively within the [range](#) **B2:B13**, provided that each individual value is strictly less than 100. This is an incredibly effective method for identifying and aggregating data points that either fail to meet an established benchmark or are categorized as 'low' based on your analytical requirements.

Imagine a typical business application where you possess a [dataset](#) documenting sales figures achieved by various employees over the course of a quarter. A manager might need to rapidly determine the total sales volume generated solely by employees who performed below a set target, such as 100 units. The following visual representation presents such a sales [dataset](#), which we will use for our practical demonstration:

	A	B	C	D	E	F
1	Employee	Sales				
2	Andy	22				
3	Bob	14				
4	Chad	10				
5	Derrick	9				
6	Eric	8				
7	Frank	11				
8	George	1459				
9	Harry	20				
10	Isaac	22				
11	John	10				
12	Kendall	15				
13	Liam	27				
14						
15						
16						
17						
18						
19						
20						
21						

To successfully sum all sales figures within the designated "Sales" column (**B2:B13**) that are strictly less than the value of 100, we must accurately implement the [SUMIF function](#) exactly as structured above.

Step-by-Step Example: Implementing "Sum If Less Than"

To execute this conditional summation in practice, first designate an empty [cell](#) within your worksheet where you intend for the calculated total to be displayed. Once the target [cell](#) is selected, carefully input the [formula](#) exactly as presented in the previous section:

=SUMIF(B2:B13, "<100")

Upon completing the [formula](#) entry, press the **Enter** key. Excel will immediately initiate the process: it scans the entirety of the specified [range](#) **B2:B13**, rigorously identifies every value that satisfies the defined [criteria](#) (being less than 100), and then accurately calculates the cumulative sum of those identified entries.

The subsequent screenshot effectively illustrates both the accurate placement of this [formula](#)

within the Excel worksheet and the resulting output generated by the function:

D2				fx		=SUMIF(B2:B13, "<100")	
	A	B	C	D		E	
1	Employee	Sales		Sum of Sales Less Than 100			
2	Andy	22		168			
3	Bob	14					
4	Chad	10					
5	Derrick	9					
6	Eric	8					
7	Frank	11					
8	George	1459					
9	Harry	20					
10	Isaac	22					
11	John	10					
12	Kendall	15					
13	Liam	27					
14							
15							
16							
17							
18							
19							
20							
21							

As clearly demonstrated, the final sum of all sales values within the designated [range](#) that strictly adhere to the "less than 100" condition is exactly **168**. This concise result offers immediate and valuable insight into the performance metrics of the lower-performing sales entries, providing a focused summary for any subsequent analysis.

Dynamic Criteria: Using Cell References with SUMIF

While manually embedding the [criteria](#) directly into the [SUMIF function](#) (e.g., using "<100") functions perfectly for static, unchanging thresholds, this method lacks the necessary flexibility for dynamic analysis. In scenarios where your comparison threshold or benchmark value is subject to frequent modifications, utilizing a [cell reference](#) as the basis for your [criteria](#) is significantly more robust and efficient. This approach allows your spreadsheets to become truly dynamic and simplifies ongoing maintenance.

To correctly integrate a [cell reference](#) into the comparison logic, you must utilize the [concatenation](#) operator (&). This operator is essential for merging the textual [comparison operator](#) (the quoted

"<" symbol) with the live numerical value stored in a specified [cell](#). For example, if your goal is to sum values in the "Sales" column (**B2:B13**) that are less than the current value residing in [cell D2](#), you would construct the following dynamic [formula](#):

=SUMIF(B2:B13, "<"&D2)

This powerful [formula](#) ensures that the threshold is read dynamically from [cell D2](#). If the content of **D2** is modified (e.g., changed from 100 to 150), the [SUMIF function](#) will instantly recalculate the sum based on the new, updated [criteria](#), entirely removing the need to manually edit the [formula](#) itself.

The subsequent screenshot effectively demonstrates this dynamic approach in a live scenario, where the value contained in [cell D2](#) is the sole factor dictating the "less than" threshold:

E2					=SUMIF(B2:B13, "<"&D2)
	A	B	C	D	E
1	Employee	Sales		Cutoff	Sum of Sales Less Than Cutoff Value
2	Andy	22		100	168
3	Bob	14			
4	Chad	10			
5	Derrick	9			
6	Eric	8			
7	Frank	11			
8	George	1459			
9	Harry	20			
10	Isaac	22			
11	John	10			
12	Kendall	15			
13	Liam	27			
14					
15					
16					
17					
18					
19					
20					
21					

Adopting this methodology significantly enhances both the adaptability and the overall maintainability of your spreadsheets, particularly when working with variable conditions or when sharing your work with colleagues who may need the ability to adjust parameters without

inadvertently modifying the underlying core [formulas](#).

Expanding Your SUMIF Capabilities: Other Comparison Operators

While the primary focus of this guide is the "less than" [criteria](#), it is important to recognize that the [SUMIF function](#) is an extremely versatile tool capable of accommodating a wide array of other [comparison operators](#). A comprehensive understanding of these operators is key to performing a broader spectrum of conditional summing tasks efficiently.

Below is a summary of the most common [comparison operators](#) that you can integrate into your **SUMIF** formulas, along with practical explanations for their usage:

"<=" (Less than or equal to): This operator is used to sum values that are less than or exactly equal to a specified number. For instance, the [formula](#) `=SUMIF(B2:B13, "<=100")` would include not only all values below 100 but also those entries that are precisely 100.

">" (Greater than): Use this operator to sum values that are strictly greater than a given numerical value. A formula like `=SUMIF(B2:B13, ">100")` would calculate the total of all sales figures that exceed 100.

">=" (Greater than or equal to): This is used to sum values that are greater than or equal to a number. Example: `=SUMIF(B2:B13, ">=100")`.

"=" (Equal to): This operator sums values that are exactly equal to a specific number or text string. While it is often optional for simple numerical comparisons (e.g., `=SUMIF(B2:B13, 100)` works), it is crucial for explicit clarity and when referencing [cells](#) or text [criteria](#) (e.g., `=SUMIF(A2:A13, "East")`).

"<>" (Not equal to): This sophisticated operator sums values that do not equal a specified number or text. For example, `=SUMIF(B2:B13, "<>100")` would aggregate all sales figures except those entries that are precisely 100.

It is imperative to maintain the rule of always enclosing the [comparison operator](#)--and the subsequent value, whether hardcoded or combined with a [cell reference](#) via the & operator--within double quotation marks in the [criteria](#) argument. This strict consistency guarantees that your formulas are correctly parsed and interpreted by [Microsoft Excel](#).

Best Practices and Troubleshooting Tips for SUMIF

To maximize the reliability of your [SUMIF function](#) formulas and ensure the long-term robustness of your spreadsheets, adhere to these key best practices and troubleshooting recommendations:

Ensure Data Type Consistency: Whenever you are performing numerical comparisons, strictly verify that the data within your [range](#) is formatted as a number. Inconsistent data types, such as numbers inadvertently stored as text, are a common cause of unexpected results or formula errors.

Mandatory Quotation Marks for Criteria: As emphasized throughout this tutorial, all [comparison operators](#) (including <, >, <=, >=, =, and <>) and any associated values must be enclosed within double quotation marks when used as the [criteria](#) argument. Text [criteria](#) must also be enclosed in quotes.

The Role of the Ampersand (&): For creating dynamic criteria, the [concatenation](#) operator (&) is absolutely vital. It is used to correctly join the quoted [comparison operator](#) with a live [cell reference](#) or the output of another [formula](#), thereby constructing the complete text string that Excel requires for evaluation.

Managing References (Absolute vs. Relative): If you anticipate copying your **SUMIF** formula across multiple [cells](#), you should utilize absolute references (e.g., \$B\$2:\$B\$13) for your [range](#) and [sum_range](#) arguments. This prevents these ranges from unexpectedly shifting when the [formula](#) is moved. Your [criteria cell reference](#) (e.g., \$D\$2) may also require the absolute reference designation depending on your design needs.

Troubleshooting Zero Results: If your **SUMIF** formula unexpectedly returns 0 when you are certain a calculated sum should exist, meticulously review your [criteria](#). Verify that there are no typographical errors, that the [range](#) is correctly specified, and critically, that the cell data types are consistent.

Additional Resources for Excel Mastery

The [SUMIF function](#) is a cornerstone of conditional aggregation, but it represents only a fraction of the powerful analytical tools available in [Microsoft Excel](#). To continue expanding your data proficiency, we highly recommend exploring these related functions and tutorials that address other essential spreadsheet challenges:

SUMIFS Function: Essential for learning how to calculate sums based on multiple simultaneous criteria.

COUNTIF/COUNTIFS Functions: Discover methods for counting the number of [cells](#) that successfully meet one or more specific conditions.

AVERAGEIF/AVERAGEIFS Functions: Understand the process of calculating averages based on one or several defined criteria.

Conditional Formatting: Explore techniques to automatically apply visual formatting (colors, icons) to [cells](#) that meet predetermined conditions, significantly enhancing data visualization.

Lookup Functions (VLOOKUP, HLOOKUP, XLOOKUP): Master the art of retrieving corresponding information from large data tables based on specific lookup values.

By consistently deepening your understanding of Excel's diverse [formulas](#) and sophisticated features, you gain the ability to convert complex data challenges into streamlined, manageable tasks, ultimately leading to more efficient and insightful data analysis.