Use SUMIF with OR in Google Sheets

Authored by

Mohammed looti

October 31, 2025

In the advanced realm of Google Sheets, performing robust conditional calculations is a fundamental requirement for effective data analysis. Analysts frequently encounter situations where they must calculate the sum of values based on one of several possible conditions being met. This requirement necessitates the implementation of an **OR logical operator** within the summation logic. While Google Sheets provides the powerful SUMIFS function, which is optimized for multiple criteria joined by AND logic, directly applying OR logic can initially appear complex. This detailed guide demonstrates the most effective and transparent technique to integrate OR conditions into your calculations, ensuring your formulas are both highly reliable and maintainable.

The SUMIFS function is inherently designed to operate under conjunctive logic; it aggregates values only when every specified criterion is simultaneously true. To introduce the required disjunctive (OR) logic, the most widely accepted and structurally sound strategy involves decomposing the OR requirement into multiple, independent SUMIFS operations. Each operation addresses one specific condition. By summing the results of these individual function calls, we effectively mimic the behavior of the OR operator. This methodology allows for precise control over which conditions contribute to the final sum, making it an indispensable tool for flexible conditional reporting and spreadsheet management.

The Strategy: Mimicking OR Logic with Multiple SUMIFS

The underlying principle for implementing OR logic in conditional summation relies on creating a separate calculation for each potential criterion and then aggregating the results. If you need to sum values where a cell meets "Criterion A" OR "Criterion B," you treat them as two distinct summation tasks. The core strength of this method lies in its ability to guarantee that any value that satisfies either condition is included in one of the individual summations, thereby correctly contributing to the final total without the risk of double-counting--assuming the criteria are mutually exclusive within the criteria range (i.e., a cell cannot be both "Value A" and "Value B" simultaneously).

This additive approach avoids the complexity and potential performance overhead associated with array formulas, offering a more explicit and easier-to-audit calculation path. When structuring the formula, the ranges to be summed and the ranges to be checked typically remain constant across all chained SUMIFS functions; only the specific criterion changes. This structural consistency further enhances the readability and maintainability of the complex formula.

Syntax and Structure of the OR Formula

To implement a two-condition OR operation, where the condition is met if a corresponding cell matches "value1" OR "value2," the syntax requires chaining two SUMIFS functions together using the addition operator (+). This structure ensures that the final output is the accumulated sum of

values that satisfy the first criterion plus the sum of values that satisfy the second criterion. The structure is highly scalable; for three or more OR conditions, one simply adds another SUMIFS block to the formula.

The standardized structure for this disjunctive operation in Google Sheets is demonstrated below, showcasing how the sum range (B2:B11) and criteria range (A2:A11) are repeated for each condition:

=SUMIFS(B2:B11, A2:A11, "value1") + SUMIFS(B2:B11, A2:A11, "value2")

In practice, the first formula segment precisely calculates the sum of values in the target range **B2:B11** where the corresponding cells in the criteria range **A2:A11** match "value1." Concurrently, the second segment performs the exact same function, but for cells matching "value2." By adding these two independent sums, we achieve the full capture of all values that meet either the "value1" or "value2" criterion. This highly versatile method remains the clearest and most accessible way to manage conditional sums involving multiple OR choices.

Case Study: Calculating Team Scores with OR Conditions

To illustrate the practical application of this technique, consider a typical scenario involving sports or financial data. Suppose you possess a dataset in Google Sheets detailing team performance and scores. Your objective is to compute the total points accumulated by two specific teams, for instance, "Mavs" or "Jazz." This task explicitly requires summing values based on an OR condition, making it an ideal candidate for the combined SUMIFS strategy.

Our spreadsheet contains team names in Column A and corresponding points in Column B. We must calculate the total points in Column B only for rows where Column A equals "Mavs" or "Jazz." The data structure is visualized below:

	A	B	C	D
1	**Team**	**Points**		
2	Mavs	23		
3	Spurs	28		
4	Rockets	18		
5	Mavs	13		
6	Mavs	19		
7	Nets	11		
8	Suns	10		
9	Spurs	14		
10	Jazz	22		
11	Rockets	34		
12				
13				
14				
15				
16				
17				

To satisfy this conditional summation requirement, we construct a formula that executes two separate conditional sums. The first sum targets the points associated with "Mavs," and the second targets the points associated with "Jazz." The summation of these two discrete results provides the required total for either team.

The resultant formula is formulated as follows:

=SUMIFS(B2:B11, A2:A11, "Mavs") + SUMIFS(B2:B11, A2:A11, "Jazz")

Upon execution, this calculation yields the aggregate points for both teams, effectively demonstrating the successful use of OR logic within the SUMIFS framework. The result, as displayed in the screenshot below, confirms that the combined points total **77**.

D2 =SUMIFS(B2:B11, A2:A11, "Mavs") + SUMIFS(B2:B11, A2:A11, "Jazz")

	A	B	C	D	E	F
1	**Team**	**Points**		**Sum of Points**		
2	Mavs	23		77		
3	Spurs	28				
4	Rockets	18				
5	Mavs	13				
6	Mavs	19				
7	Nets	11				
8	Suns	10				
9	Spurs	14				
10	Jazz	22				
11	Rockets	34				
12						
13						
14						
15						
16						
17						

Ensuring Accuracy: Verification and Debugging

A crucial phase in any data analysis workflow is the verification of calculated results. To ensure the absolute accuracy of our combined formula, especially when dealing with OR conditions, a manual check against the raw data is highly recommended. This step solidifies confidence that the SUMIFS function correctly identified and aggregated all values meeting the specified criterion.

We proceed by manually reviewing the "Points" column for rows corresponding to either "Mavs" or "Jazz" in the "Team" column. The visual inspection helps confirm which rows were included in the calculation:

	A	B	C	D	E
1	**Team**	**Points**		**Sum of Points**	
2	Mavs	23		77	
3	Spurs	28			
4	Rockets	18			
5	Mavs	13			
6	Mavs	19			
7	Nets	11			
8	Suns	10			
9	Spurs	14			
10	Jazz	22			
11	Rockets	34			
12					
13					
14					
15					
16					

From the source data, we can isolate the contributing points: the scores for "Mavs" are 23 and 19. The scores for "Jazz" are 13 and 22. Summing these manually yields: 23 + 13 + 19 + 22 = **77**. This manual summation precisely matches the result generated by our chained formula. This verification process is fundamental to maintaining data integrity and building trust in complex spreadsheet calculations.

Advanced Considerations and Best Practices

While the methodology of adding multiple SUMIFS statements is highly effective for OR logic, adopting certain best practices ensures superior spreadsheet management, greater flexibility, and enhanced readability across large datasets.

Dynamic Criteria Referencing: Instead of embedding specific text strings like "Mavs" or "Jazz" directly into the formula (hardcoding), it is best practice to reference cells containing these criteria. If, for instance, the criteria are placed in cells D1 and D2, the formula becomes dynamic: `=SUMIFS(B2:B11, A2:A11, D1) + SUMIFS(B2:B11, A2:A11, D2)`. This allows users to change the targets instantly without touching the formula itself.

Using Named Ranges: For significantly improved clarity and maintainability, define named ranges for your data blocks. Naming **B2:B11** as "Points" and **A2:A11** as "Teams" transforms the formula into `=SUMIFS(Points, Teams, "Mavs") + SUMIFS(Points, Teams, "Jazz")`, making the intent of the calculation immediately obvious to any user.

Handling Extensive Criteria Lists: If the list of criteria grows very large, chaining dozens of SUMIFS functions can become cumbersome. For such extended disjunctive requirements, more advanced alternatives like the `QUERY` function or a combination of `SUM` and `ARRAYFORMULA` with `REGEXMATCH` should be considered. However, for a moderate number of OR conditions, the technique described here remains the most straightforward and easiest to debug.
Distinguishing SUMIF vs. SUMIFS: While SUMIF handles a single criterion, SUMIFS is generally preferred even for single criteria because its argument order (sum range first) is consistent and more flexible for incorporating secondary AND conditions if needed alongside the primary OR structure.

Conclusion and Further Exploration

Mastering conditional summation in Google Sheets, particularly when dealing with OR conditions, is critical for precise data analysis. By employing the technique of combining multiple SUMIFS functions, you can accurately aggregate data across various disjunctive requirements with clarity and efficiency. Always remember to verify your formulas and utilize best practices, such as cell references and named ranges, to ensure the robustness and longevity of your spreadsheet models.

For a comprehensive understanding of the SUMIFS function and its official parameters, you can consult the official Google Sheets documentation.

Explore further tutorials to deepen your expertise in Google Sheets formulas:

How to Use SUMIF with Dates in Google Sheets
How to Use SUMIF with Multiple Columns in Google Sheets
How to Use SUMIF with NOT EQUAL To in Google Sheets