

Sum by Month in Google Sheets (With Example)

Authored by
Mohammed loot

October 29, 2025

RECOMMENDED CITATION

Mohammed loot (2025). *Sum by Month in Google Sheets (With Example)*.
PSYCHOLOGICAL STATISTICS. Retrieved from
<https://statistics.arabpsychology.com/?p=5259>

In the realm of [business intelligence](#) and financial reporting, the ability to aggregate transactional data based on time periods is indispensable. A frequent requirement involves performing [data aggregation](#) to **sum values by month** within large data sets. This powerful capability allows users of [Google Sheets](#) to transform raw, granular entries into concise, actionable summaries, providing essential insights into trends, seasonal performance, and key performance indicators (KPIs).

Whether you are managing complex inventory records, auditing detailed expense reports, or analyzing customer retention rates, monthly summaries offer a vital level of abstraction necessary for effective decision-making. This comprehensive tutorial will guide you through a meticulous, four-step methodology using native Google Sheets functions. By the end of this guide, you will be equipped to efficiently organize, calculate, and present your monthly data totals with accuracy and clarity.

Step 1: Laying the Foundation with Structured Data

The success of any sophisticated data manipulation task hinges entirely on the initial preparation and structure of the raw input. To accurately calculate monthly totals, your [dataset](#) must be organized into a format that the spreadsheet software can interpret logically. For this specific task, a minimum of two columns is required: one dedicated to the event **date** and another containing the corresponding **value** (the numerical quantity you intend to sum).

Begin by ensuring your data is entered into [Google Sheets](#), guaranteeing that the dates are consistently formatted. Inconsistent date formats (e.g., mixing US `MM/DD/YYYY` with European `DD/MM/YYYY`) can lead to catastrophic calculation errors when the software attempts to extract the month. A correctly structured dataset forms the bedrock of [structured data](#) analysis, making the subsequent formula application seamless and reliable.

For our practical demonstration, we are utilizing columns A and B. Column A holds the transaction dates, and column B contains the associated sales figures. This setup is critical, as it defines the two primary ranges we will reference throughout the monthly summing process: the criterion range (dates/months) and the sum range (values).

	A	B	C	D
1	Date	Sales		
2	1/1/2022	40		
3	1/19/2022	45		
4	1/24/2022	32		
5	2/2/2022	38		
6	2/13/2022	12		
7	3/14/2022	40		
8	3/15/2022	22		
9	3/17/2022	24		
10	3/20/2022	25		
11				
12				
13				
14				
15				
16				
17				
18				

Step 2: Isolating the Monthly Identifier Using the MONTH() Function

Since the `SUMIF()` function cannot directly group based on the month within a full date string, the next crucial step is to create a helper column that extracts the month number from each date entry. This extraction is essential because it provides a consistent, numerical grouping variable (1 through 12) that we can use as a [criterion](#) for our summation. This conversion transforms a time-specific entry into a category identifier.

[Google Sheets](#) simplifies this process with the built-in [MONTH\(\) function](#). This function accepts a date value as its sole argument and returns the month as an integer. For instance, if the date is 2024-03-15, the function returns 3, representing March. This numerical output is perfectly suited for array operations and conditional summing.

To implement this extraction, select an empty column adjacent to your data (we use Column D in this example). In the second row (cell **D2**), input the following formula, referencing the first date in your date column (cell **A2**):

=MONTH(A2)

After confirming the formula with Enter, cell D2 will display the corresponding month number. To populate this formula down the entire [dataset](#), utilize the fill handle--the small square located at the bottom-right corner of cell D2. Dragging this handle downwards automatically adjusts the cell references (A2 becomes A3, A4, and so forth), ensuring that column D accurately reflects the month number for every date entry in column A.

	A	B	C	D	E
1	Date	Sales		Month	
2	1/1/2022	40		1	
3	1/19/2022	45		1	
4	1/24/2022	32		1	
5	2/2/2022	38		2	
6	2/13/2022	12		2	
7	3/14/2022	40		3	
8	3/15/2022	22		3	
9	3/17/2022	24		3	
10	3/20/2022	25		3	
11					
12					
13					
14					
15					
16					
17					
18					

Step 3: Generating a Unique List of Monthly Criteria

With the month numbers successfully isolated in Column D, our next objective is to define the specific categories against which we will calculate the totals. If our raw data spans several months, we need a clean, non-repetitive list of those months. This list will serve as the [criterion](#) input for our final summation function, ensuring that we calculate a total only once for each unique month present in the source data.

[Google Sheets](#) streamlines this process using the highly effective [UNIQUE\(\) function](#). The primary role of the [UNIQUE\(\) function](#) is to process a specified range of data and return only the distinct values, automatically eliminating all duplicates. This functionality is ideal for generating a reliable list of monthly identifiers.

Select an empty cell far enough from your main data to prevent overwriting (cell **F2** in our scenario) and enter the following formula, referencing the entire range of month numbers created in the previous step (Column D):

=UNIQUE(D2:D10)

Upon execution, the [UNIQUE\(\) function](#) will automatically populate cells F2 and downwards with the unique month numbers (e.g., 1, 2, 3), creating a dynamic array. This dynamically generated list is the most efficient way to establish the grouping mechanism for the subsequent summation step, ensuring every existing month is accounted for exactly once.

	A	B	C	D	E	F
F2						=UNIQUE(D2:D10)
1	Date	Sales		Month		Unique Months
2	1/1/2022	40		1		1
3	1/19/2022	45		1		2
4	1/24/2022	32		1		3
5	2/2/2022	38		2		
6	2/13/2022	12		2		
7	3/14/2022	40		3		
8	3/15/2022	22		3		
9	3/17/2022	24		3		
10	3/20/2022	25		3		
11						
12						
13						
14						
15						
16						
17						

Step 4: Calculating Monthly Aggregates with SUMIF()

The final and most crucial stage is applying conditional summation to aggregate the values based on the unique monthly identifiers we just generated. This is accomplished using the robust [SUMIF\(\) function](#), a powerful tool designed to sum values in a range only if those values meet a specific condition or [criterion](#).

Understanding the [syntax](#) of the [SUMIF\(\) function](#) is key: `SUMIF(range, criterion, sum_range)`. Each component plays a distinct role:

range (Criterion Range): This is the column containing the monthly identifiers (Column D: `D2:D10`). This range is where the function looks for the match.

criterion: This is the specific month number (e.g., 1, 2, or 3) we are currently summing. This should be a relative reference to the unique month list (e.g., `F2`).

sum_range: This is the column containing the actual numerical values (sales figures) to be summed (Column B: `B2:B10`).

It is paramount to utilize [absolute references](#) (using dollar signs, e.g., `D2:D10`) for both the `range` and the `sum_range` arguments. This practice ensures that when you copy the formula down to calculate subsequent months, the source data ranges remain fixed, preventing incorrect calculations caused by shifting boundaries.

Enter the following formula into cell **G2**, which is adjacent to the first unique month identifier (F2):

`=SUMIF(D2:D10, F2, B2:B10)`

After pressing Enter, cell G2 will immediately display the total aggregated sales for the month represented by the value in F2 (January). To complete the analysis, simply drag the fill handle from cell G2 down through the rest of the rows corresponding to your list of unique months in Column F. This action will instantaneously populate Column G with the accurate monthly totals, providing the desired summary.

G2 fx =SUMIF(\$D\$2:\$D\$10, F2, \$B\$2:\$B\$10)

	A	B	C	D	E	F	G
1	Date	Sales		Month		Unique Months	Total Sales
2	1/1/2022	40		1		1	117
3	1/19/2022	45		1		2	50
4	1/24/2022	32		1		3	111
5	2/2/2022	38		2			
6	2/13/2022	12		2			
7	3/14/2022	40		3			
8	3/15/2022	22		3			
9	3/17/2022	24		3			
10	3/20/2022	25		3			
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							
21							

This final, organized output provides immediate and clear [data analysis](#), converting hundreds of individual transactions into a manageable, summary report. Based on the illustration, the summary reveals:

Total sales recorded in **January** (Month 1) totaled **117** units.

Total sales recorded in **February** (Month 2) totaled **50** units.

Total sales recorded in **March** (Month 3) totaled **111** units.

This systematic approach using the [MONTH\(\) function](#), the [UNIQUE\(\) function](#), and the [SUMIF\(\) function](#) effectively transforms raw transactional inputs into meaningful, structured monthly summaries.

Conclusion

The ability to efficiently sum values by month is a cornerstone skill for anyone utilizing [Google Sheets](#) for reporting or [data analysis](#). By mastering the four-step workflow--data preparation, month extraction, unique criteria generation, and conditional summation--you can quickly transform dense, time-series data into clear, actionable monthly insights. This technique ensures data integrity while providing the necessary high-level overview required for strategic planning and performance evaluation. Proficiency in these functions significantly enhances your capacity to

manage and interpret complex [spreadsheet software](#) data.

Additional Resources

To further enhance your spreadsheet proficiency, explore related tutorials that cover advanced data manipulation techniques: