How to Sum Values by Week in Google Sheets

Authored by

stats writer

January 16, 2026

To calculate aggregated totals by week in Google Sheets, a combination of powerful functions is required to first standardize the time component and then perform conditional summation. The primary approach involves using the **WEEKNUM function** to assign a standardized week number to every date in your dataset. Once the dates are categorized, you can employ the **SUMIF** or **SUMIFS** function to efficiently sum up all associated numerical values corresponding to that specific week identifier. This systematic method allows for precise weekly performance tracking and detailed trend analysis across your entire dataset.

While the function-based method provides granular control over criteria and output location, an alternative, often faster method involves leveraging the built-in **Pivot Table** functionality in Google Sheets. Pivot tables can automatically group date-based data by week, quarter, or year, simplifying the process of calculating sums for each defined period. This tutorial will walk you through the manual, function-based technique, offering a deep dive into the powerful formulas used for complex data aggregation tasks.

Analyzing performance or activity based on weekly intervals is a fundamental requirement for many business and analytical tasks within Google Sheets. Whether you are tracking sales, operational metrics, or project progress, summing data by week provides essential insights into short-term fluctuations and long-term momentum.

The following example provides a detailed, step-by-step guide on how to implement this calculation effectively using native Google Sheets functions.

Step 1: Preparing Your Dataset in Google Sheets

The first critical step in performing weekly data summation is ensuring your raw data is correctly formatted and structured. For this methodology to work, you must have at least two columns: one column dedicated strictly to dates and another column containing the numerical values you intend to sum (e.g., sales figures, clicks, or expenses). Consistency in date formatting is key to avoiding errors when using time-based functions.

We begin by inputting the dataset. For this illustration, we will track daily sales figures for a product across a specific timeframe. Ensure your dates are recognizable as date formats by Google Sheets, otherwise, the **WEEKNUM** function will not execute correctly. Dates should typically reside in column A and the corresponding numerical values (Sales) in column B.

This setup establishes the foundation for our analysis, linking each recorded value to a specific point in time. Reviewing the initial data structure helps verify that the subsequent steps will operate on clean and reliable inputs. The structure shown below represents a typical layout for time-series data requiring weekly summarization:

	A	B	C	D
1	**Date**	**Sales**		
2	1/1/2023	13		
3	1/3/2023	9		
4	1/9/2023	14		
5	1/13/2023	10		
6	1/15/2023	22		
7	1/16/2023	23		
8	1/20/2023	18		
9	2/3/2023	12		
10	2/5/2023	10		
11	2/16/2023	8		
12	2/17/2023	5		
13				
14				
15				
16				

Step 2: Utilizing the WEEKNUM Function for Standardization

Since dates are continuous and granular, we must transform them into discrete, measurable units--in this case, week numbers. The **WEEKNUM function** is specifically designed for this purpose, extracting the chronological week number (1 to 52 or 53) from a given date. This standardization is vital because it provides the common criterion needed later for the conditional summation.

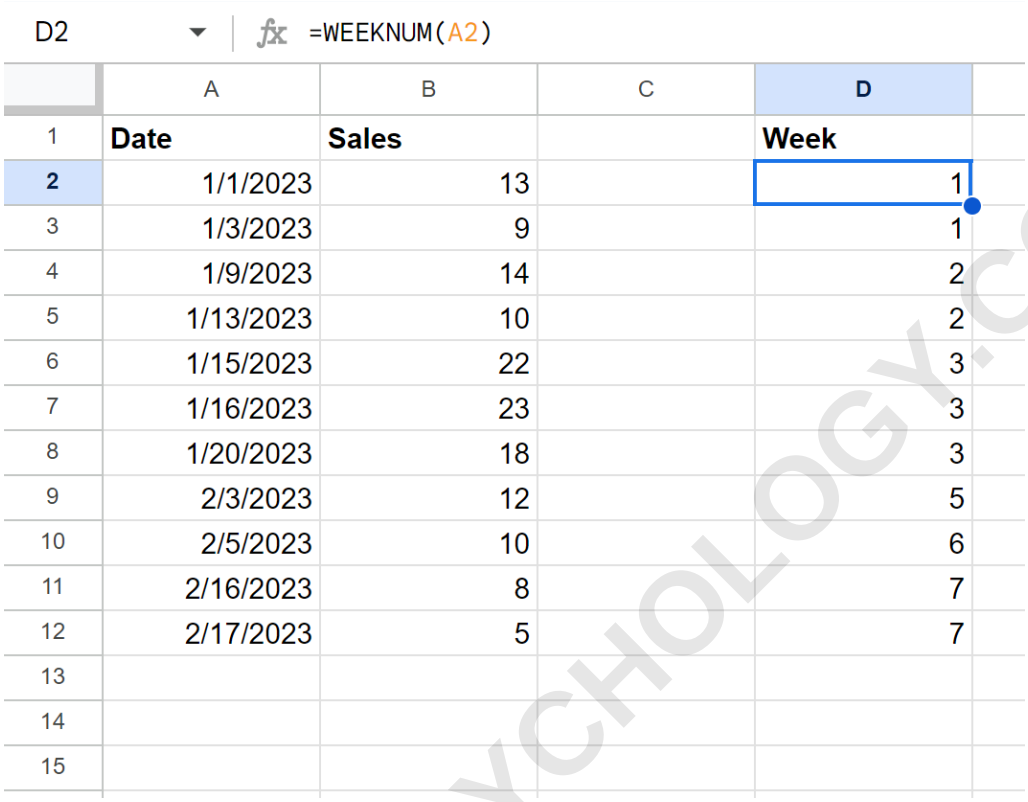
The syntax for the WEEKNUM function is straightforward: `=WEEKNUM(date, )`. The optional argument specifies which day the week starts on (e.g., 1 for Sunday, 2 for Monday). While the default setting often suffices, defining the type ensures consistency, especially in international datasets where week start conventions may vary.

In our example, we will add a new column, Column D, labeled "Week Number." We will enter the formula to calculate the week number based on the date found in cell **A2**. We use the most basic form of the function, assuming the default week start convention:

=WEEKNUM(A2)

After entering this formula in cell **D2**, it is crucial to apply it to all remaining rows in the dataset. This is accomplished by using the fill handle (the small square at the bottom right corner of the selected

cell) and dragging the formula down to the last row containing data. This action automatically adjusts the cell reference (A2 becomes A3, A4, and so on) for each subsequent date, populating Column D with the corresponding week number for every entry. This result provides the necessary grouping variable for the next steps.



	A	B	C	D
1	**Date**	**Sales**		**Week**
2	1/1/2023	13		1
3	1/3/2023	9		1
4	1/9/2023	14		2
5	1/13/2023	10		2
6	1/15/2023	22		3
7	1/16/2023	23		3
8	1/20/2023	18		3
9	2/3/2023	12		5
10	2/5/2023	10		6
11	2/16/2023	8		7
12	2/17/2023	5		7
13				
14				
15				

Step 3: Isolating Unique Week Identifiers Using the UNIQUE Function

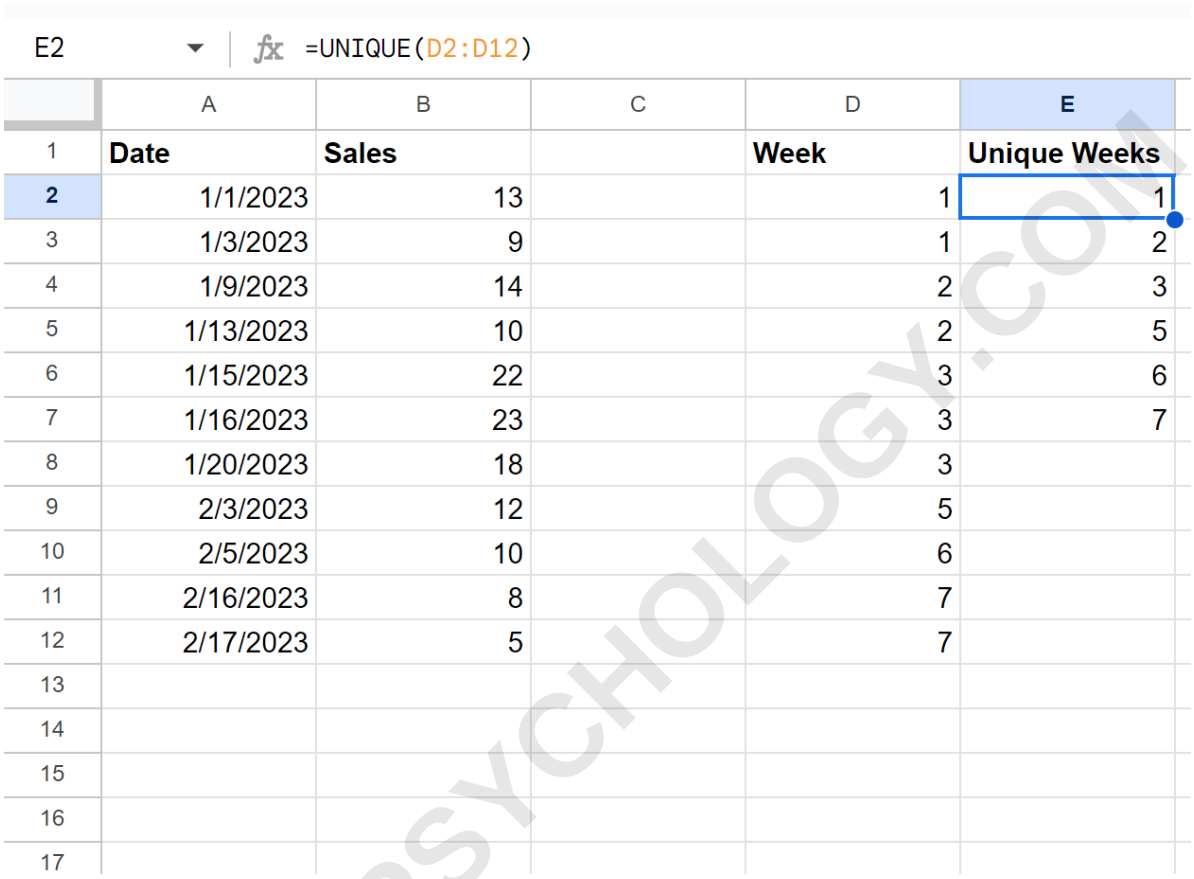
Once Column D is populated with week numbers, we face a slight challenge: to sum the data, we need a distinct list of the weeks present in the dataset so we can iterate through them. If we try to sum against the entire column of repetitive week numbers, our results will be scattered and difficult to manage.

The **UNIQUE function** solves this by automatically extracting a list of every distinct value found within a specified range. This means that if Week 1 appears ten times in Column D, the UNIQUE function will return "1" only once, creating a clean list of all weeks that require summation.

We will place this output in a new location, perhaps starting in cell **E2**. The formula specifies the range containing the week numbers we generated in the previous step (Column D). In this particular example, the range spans from D2 to D12:

=UNIQUE(D2:D12)

Upon execution, the UNIQUE function dynamically spills the unique week numbers into Column E. This list serves as the definitive set of criteria against which we will perform our conditional summing. This dynamic list simplifies the subsequent calculation step significantly, as we only need to write the summation formula once and drag it down alongside these unique identifiers.



	A	B	C	D	E
1	**Date**	**Sales**		**Week**	**Unique Weeks**
2	1/1/2023	13		1	1
3	1/3/2023	9		1	2
4	1/9/2023	14		2	3
5	1/13/2023	10		2	5
6	1/15/2023	22		3	6
7	1/16/2023	23		3	7
8	1/20/2023	18		3	
9	2/3/2023	12		5	
10	2/5/2023	10		6	
11	2/16/2023	8		7	
12	2/17/2023	5		7	
13					
14					
15					
16					
17					

Step 4: Applying the SUMIF Function for Weekly Totals

With our unique week identifiers established in Column E, we can now calculate the total sales corresponding to each week using the powerful **SUMIF function**. The SUMIF function is designed to sum values in a range based on a single condition.

The general syntax for **SUMIF** is `=SUMIF(range, criterion, sum_range)`:

`range`: This is the range where the criteria will be checked--in our case, the column containing all the week numbers (Column D).

`criterion`: This is the condition that must be met--the specific unique week number we are currently summing for (found in Column E).

`sum_range`: This is the range containing the actual values to be summed (Column B, the Sales

figures).

We will enter the formula in cell **F2**, adjacent to the first unique week number in E2. It is absolutely essential to use absolute references (using the **$** sign) for both the `range` and `sum_range` (Columns D and B) to ensure that these ranges do not shift when the formula is dragged down. The `criterion` reference (E2) must remain relative so it correctly checks against each unique week number.

The resulting formula is:

=SUMIF($D$2:$D$12, E2, $B$2:$B$12)

After calculating the sum for the first week in cell F2, drag the fill handle down to apply the formula to the remaining unique week identifiers listed in Column E. This final action completes the weekly data aggregation process, providing a summary table that clearly delineates the total sales achieved during each specific week of the recorded period.

F2 =SUMIF($D$2:$D$12, E2, $B$2:$B$12)

	A	B	C	D	E	F
1	**Date**	**Sales**		**Week**	**Unique Weeks**	**Sum of Sales**
2	1/1/2023	13		1	1	22
3	1/3/2023	9		1	2	24
4	1/9/2023	14		2	3	63
5	1/13/2023	10		2	5	12
6	1/15/2023	22		3	6	10
7	1/16/2023	23		3	7	13
8	1/20/2023	18		3		
9	2/3/2023	12		5		
10	2/5/2023	10		6		
11	2/16/2023	8		7		
12	2/17/2023	5		7		
13						
14						
15						

Step 5: Reviewing and Interpreting the Results

The resulting output in Columns E and F provides a clean, summarized view of the data, grouped precisely by the week number. This aggregation transforms raw, daily transactional data into actionable weekly metrics, essential for performance review and decision-making. By comparing

the totals across different weeks, one can quickly identify peak periods, low points, and overall trends.

For instance, reviewing the summary table reveals specific weekly performance metrics. This allows analysts to quantify success or failure within defined time buckets. The interpretation of these weekly sums is often the most important step, as it connects the numerical results back to real-world business outcomes.

From the structured summary output, we can draw immediate, quantifiable conclusions:

There were **22** total sales made in the first week of the year, establishing a baseline performance.

Total sales rose slightly to **24** during the second week, indicating minor growth or stability.

A significant increase to **63** total sales was observed in the third week of the year, highlighting a potential promotional success or a strong cyclical demand period.

Understanding these variances helps stakeholders adjust inventory, allocate marketing resources, or investigate the underlying causes of sudden spikes or drops in activity. The clarity provided by the weekly sum is far superior to analyzing the raw, unsorted daily data.

Alternative Method: Leveraging Pivot Tables for Date Grouping

While the function-based approach offers maximum transparency and flexibility, for users seeking a quicker, less formula-intensive method, the **Pivot Table** feature in Google Sheets provides an excellent alternative for date-based data aggregation.

A Pivot Table allows you to summarize large datasets by rearranging (pivoting) the data based on various dimensions. When using date fields, Google Sheets automatically provides options for grouping. To use this method, select your entire dataset (Columns A and B in our example) and insert a new Pivot Table.

In the Pivot Table editor, add the Date column to the "Rows" area. Crucially, right-click on the date values within the pivot table itself and select "Create date group." Google Sheets will present options such as Year, Quarter, Month, and Week. Selecting the "Week of year" option will automatically group all entries under their respective week numbers.

Next, move the numerical column (Sales) to the "Values" area, ensuring the calculation selected is "SUM." The resulting pivot table will instantly display the total sum of sales for every distinct week in the data, accomplishing the same task as the **WEEKNUM** and **SUMIF** combination, often with fewer steps and less risk of formula error. This method is highly recommended for ad-hoc analysis or when the output location doesn't need to be integrated into an existing spreadsheet structure.

Conclusion: Enhancing Data Analysis Capabilities

Mastering weekly aggregation in Google Sheets, whether through the precision of the **WEEKNUM** and SUMIF function combination or the efficiency of the **Pivot Table** tool, significantly enhances your capacity for data analysis. These techniques transform raw daily entries into meaningful weekly summaries, providing the temporal context necessary for effective business intelligence.

Choosing the right method depends on the context: the formula-based approach offers reusable structure and is ideal for dashboards or reports requiring data integration, while the Pivot Table is superior for exploratory analysis and quick summarization. Regardless of the choice, the ability to rapidly calculate and interpret weekly totals is a core skill for any advanced user working with time-series data.

The following tutorials explain how to perform other common tasks in Google Sheets:
How to Calculate Year to Date Values in Google Sheets