

How to Find the First Monday of Each Month in Google Sheets

Authored by
stats writer

February 7, 2026

RECOMMENDED CITATION

stats writer (2026). *How to Find the First Monday of Each Month in Google Sheets*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=129610>

Google Sheets is renowned as an extremely robust and versatile spreadsheet application, particularly adept at handling complex data manipulation, scheduling, and temporal analysis. A frequent requirement in professional settings is the need to pinpoint specific cyclical dates, such as the exact date of the **first Monday** of any given month. While manually checking a calendar for individual dates is feasible, automating this process across large datasets is essential for efficient project management, payroll processing, or recurring event planning. This comprehensive guide details the mathematical logic and the exact formula required to achieve this goal seamlessly within the Google Sheets environment.

By leveraging a combination of built-in date functions, users can construct a single, powerful expression that automatically identifies the required date, eliminating the potential for human error associated with manual tracking. This technique demonstrates the power of combining simple functions like DATE, YEAR, and MONTH with the specialized temporal calculation capabilities of the **WEEKDAY** function. Understanding the nuances of this solution allows spreadsheet users to elevate their data analysis skills and create highly dynamic, self-updating spreadsheets for various organizational needs.

The ability to dynamically calculate the first Monday is invaluable for organizational efficiency. Imagine managing a portfolio of projects where status meetings must always fall on the first Monday of the reporting cycle, or managing a budget where financial reports are due on that exact day. Instead of adjusting dates manually every month, a single formula handles the complexity, providing a convenient and highly reliable solution. We will now introduce the core formula and then proceed to break down its components rigorously.

Google Sheets: Finding the First Monday of Each Month

The Essential Formula: Locating the First Monday

The calculation to determine the first Monday of any given month involves a carefully constructed mathematical relationship between the date of the seventh day of the month and its position within the week. By establishing the 7th as an anchor point, we guarantee that the preceding Monday falls within the first seven days of the month, thus guaranteeing it is the first occurrence.

The following formula, used within Google Sheets, successfully isolates the first Monday based on any input date found in the referenced cell. Note that the reference point, **A2**, can contain any date within the month you wish to analyze; the formula extracts only the year and month information for its calculation.

```
=DATE(YEAR(A2),MONTH(A2),7)-WEEKDAY(DATE(YEAR(A2),MONTH(A2),7),3)
```

In this particular presentation, the formula is designed to return the exact date of the **first Monday** corresponding to the month and year derived from the date currently stored in cell A2. The resulting output will be a standard date serial number, which Google Sheets automatically formats into a readable date format based on your locale settings. This single line of code efficiently solves a complex calendar problem.

Before diving into the detailed breakdown of how each function interacts, we will first walk through a practical example demonstrating the setup and execution of this formula within a typical spreadsheet context. This hands-on visualization will solidify the understanding of where and how this powerful calculation should be applied to your data sets.

Prerequisites and Setup: Preparing Your Data

To follow the forthcoming example, it is assumed that you have a column populated with various dates. These dates serve as the input references from which the formula will extract the required month and year information. It is crucial that these input cells contain valid date formats for the YEAR and MONTH functions to operate correctly.

Suppose, for instance, that we have an active dataset listing various report starting dates, indexed chronologically in Column A. The goal is to create a parallel column (Column B) that identifies the corresponding **first Monday** for each of these months. This structure is common in scheduling matrices or fiscal period management. The following illustration shows a hypothetical list of initial dates in the spreadsheet:

	A	B	C	D
1	Date			
2	1/1/2023			
3	2/1/2023			
4	3/1/2023			
5	4/1/2023			
6	5/1/2023			
7	6/1/2023			
8	7/1/2023			
9	8/1/2023			
10	9/1/2023			
11	10/1/2023			
12	11/1/2023			
13	12/1/2023			
14				
15				
16				

The dates listed in Column A--starting with **1/1/2023** in cell A2--will serve as the basis for our calculation. The power of this approach lies in the fact that the actual day of the month (the '1' in 1/1/2023) is irrelevant; only the month (January) and the year (2023) are extracted for use in the formula. This preparation ensures that our calculation is robust and unaffected by the starting date's position within its week.

Step-by-Step Implementation Example

We begin the implementation process by entering the complete formula into the first corresponding target cell, which is **B2**. This calculation will analyze the date provided in **A2** (1/1/2023) and return the first Monday of January 2023. The formula, referencing A2, must be typed exactly as follows:

=DATE(YEAR(A2),MONTH(A2),7)-WEEKDAY(DATE(YEAR(A2),MONTH(A2),7),3)

Upon pressing Enter, Google Sheets will execute the calculation and display the resulting date in **B2**. The next critical step is applying this formula efficiently to the rest of the dataset in Column A. Instead of manually retyping the formula for every row, we utilize the standard spreadsheet functionality of "click and drag" or the fill handle.

By selecting cell B2 and dragging the formula downward along Column B, the relative cell

references (A2, A3, A4, etc.) automatically adjust. This allows the same calculation to be performed for every date listed in Column A, immediately populating Column B with the desired first Monday dates, as shown in the following visual representation of the completed process:

B2 fx =DATE(YEAR(A2),MONTH(A2),7)-WEEKDAY(DATE(YEAR(A2),MONTH(A2),7),3)

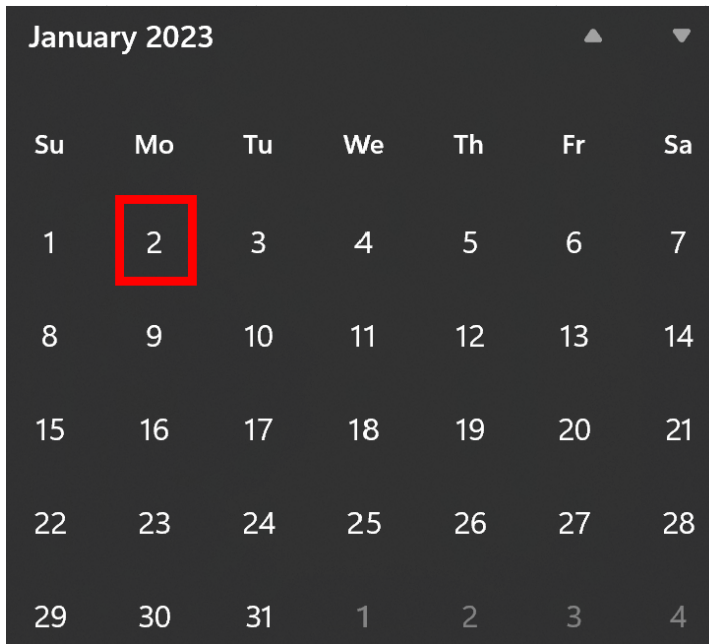
	A	B	C	D	E	F
1	Date	First Monday				
2	1/1/2023	1/2/2023				
3	2/1/2023	2/6/2023				
4	3/1/2023	3/6/2023				
5	4/1/2023	4/3/2023				
6	5/1/2023	5/1/2023				
7	6/1/2023	6/5/2023				
8	7/1/2023	7/3/2023				
9	8/1/2023	8/7/2023				
10	9/1/2023	9/4/2023				
11	10/1/2023	10/2/2023				
12	11/1/2023	11/6/2023				
13	12/1/2023	12/4/2023				
14						
15						
16						

As illustrated, Column B now accurately contains the calculated date of the **first Monday** corresponding to the month indicated in the adjacent cell of Column A. This mass application demonstrates the efficiency of the formula, transforming a tedious manual task into a quick, automated process.

Verification and Cross-Referencing

To ensure the accuracy of the complex formula, it is necessary to verify the results against a reliable external source, such as a traditional calendar. Our first input date, **1/1/2023**, corresponds to January 2023. The formula successfully returned **1/2/2023** as the first Monday of that month.

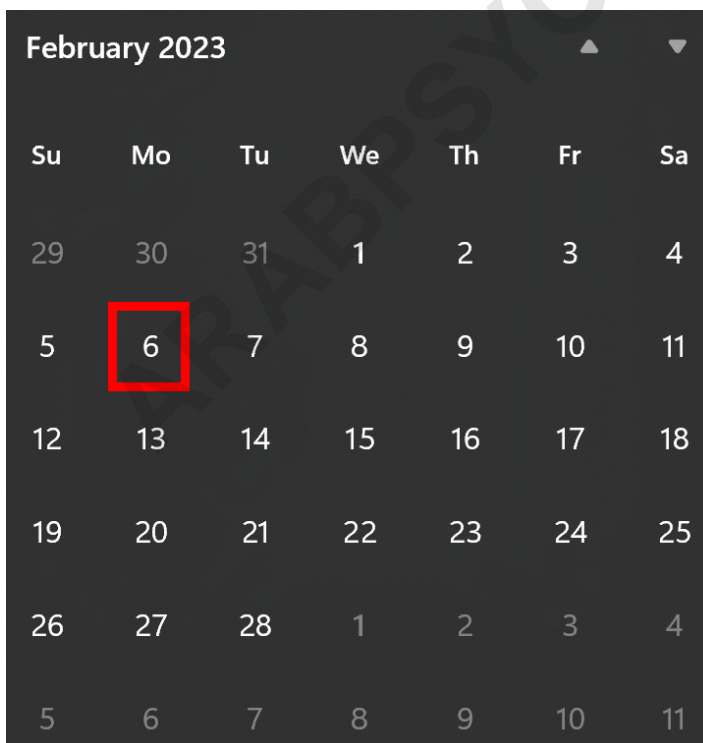
A quick reference to the calendar for January 2023 confirms this finding. Since January 1st fell on a Sunday, the next day, January 2nd, was indeed the first Monday of the month. This initial verification step is crucial for establishing confidence in the automated calculation:



January 2023

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Similarly, we can examine the second date in our list, **2/1/2023**, which corresponds to February 2023. The calculated result in Column B is **2/6/2023**. Referring once again to the calendar, we find that February 1st, 2023, was a Wednesday. Counting forward, the first Monday in that month aligns perfectly with the formula's output:



February 2023

Su	Mo	Tu	We	Th	Fr	Sa
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	1	2	3	4
5	6	7	8	9	10	11

These confirmations underscore the precision of the logic embedded within the formula.

Regardless of the starting day of the month, the calculation consistently isolates the correct Monday, making the entire system highly dependable for time-sensitive scheduling tasks across different years and months.

Dissecting the Logic: How the Formula Functions

To fully appreciate the elegance and reliability of this solution, we must analyze the formula piece by piece. The structure, which references the source date in cell A2, is designed to perform two main actions: first, it establishes a reliable anchor date within the target month, and second, it calculates the offset required to reach the preceding Monday.

=DATE(YEAR(A2),MONTH(A2),7)-WEEKDAY(DATE(YEAR(A2),MONTH(A2),7),3)

The first major component is the anchor date calculation: `DATE(YEAR(A2),MONTH(A2),7)`. This segment utilizes the YEAR and MONTH functions to extract the year and month from the initial date in **A2**, respectively. Crucially, the day argument is set to **7**. By selecting the seventh day of the month, we guarantee that this date falls within the first week of that month, regardless of whether the month starts on a Sunday or a Saturday. For the initial example date of 1/1/2023, this entire section returns **1/7/2023**.

The second component involves the **WEEKDAY** calculation, which determines how far the anchor date (the 7th) is from a Monday. This is where the specific return type argument becomes vital. The syntax is `WEEKDAY(DATE(YEAR(A2),MONTH(A2),7),3)`. The optional argument **3** instructs the WEEKDAY function to return a number where Monday is represented by 0, Tuesday by 1, and so on, up to Sunday being 6. This specific numbering scheme simplifies the final subtraction. Using the example of 1/7/2023 (a Saturday), the WEEKDAY function returns **5**, since Saturday is five positions away from Monday (0).

Lastly, the subtraction sign (the main operator connecting the two segments) takes the anchor date (1/7/2023) and subtracts the result of the **WEEKDAY** calculation (5). In date arithmetic, subtracting the number 5 is equivalent to rolling the date back five days. Subtracting 5 days from 1/7/2023 results in **1/2/2023**. This method ensures that we land exactly on the preceding Monday, which, because we anchored to the 7th, must be the first Monday of the month. This process is repeated row by row, ensuring precise calculation across the entire dataset.

Conclusion: Automating Scheduling and Planning

The ability to accurately and automatically locate specific days of the week, such as the **first Monday**, within a spreadsheet environment transforms Google Sheets from a simple data repository into a powerful scheduling and analytical engine. By implementing the detailed formula

discussed here, users gain a reliable tool for automating recurring administrative tasks and improving temporal planning accuracy.

This method, relying on the robust date functions, is highly scalable and requires minimal maintenance once implemented. Whether you are managing complex financial reporting deadlines or coordinating international project milestones, utilizing this specific combination of the WEEKDAY and DATE functions ensures that your scheduling is always anchored to the correct, predictable starting point of the month.

For those interested in further maximizing their productivity with Google Sheets, exploring other common date and time manipulation tutorials can be highly beneficial. Mastering functions related to calculating working days, determining month-end dates, or handling time zone conversions are logical next steps in building advanced data management capabilities.

The following tutorials explain how to perform other common tasks in Google Sheets: