

# Learning to Determine the First Day of the Week in Google Sheets

Authored by  
**Mohammed loot**

November 14, 2025

## RECOMMENDED CITATION

Mohammed loot (2025). *Learning to Determine the First Day of the Week in Google Sheets*. PSYCHOLOGICAL STATISTICS. Retrieved from <https://statistics.arabpsychology.com/?p=934>

## Introduction: Navigating Regional Week Start Conventions

For professionals engaged in data management, accurately parsing and manipulating chronological data is essential. A frequent requirement in [Google Sheets](#) involves precisely determining the first [day of the week](#) associated with any given [date](#). This task is complicated by significant regional variations in defining the week's commencement. While the North American convention traditionally establishes **Sunday** as the inaugural day, global standards--specifically those adhering to the widely recognized [ISO 8601 standard](#)--mandate **Monday** as the official start. Recognizing and accommodating these differing standards is crucial for generating accurate reports, ensuring compliance, and performing reliable calculations within your [spreadsheet](#) environment.

This definitive tutorial provides highly robust [formulas](#) designed specifically for [Google Sheets](#). These mathematical expressions successfully identify the start date of the week, whether your operational standard dictates a Sunday or a Monday start. We will meticulously break down the underlying logic, demonstrating how these calculations function regardless of the input date. Furthermore, we will illustrate the practical implementation through clear, sequential steps. By mastering these specialized date handling techniques, you will significantly elevate your data analysis capabilities, producing results that are both mathematically sound and contextually appropriate for your specific regional or international reporting needs.

Our primary methodology centers on the powerful native capabilities of [Google Sheets](#), with a focus on leveraging the [MOD function](#) (the modulo operation). This function is indispensable for calculating the exact numerical offset required to backtrack from any date to the beginning of its respective weekly cycle. We will guide you through the process, which includes setting up your source data, precisely applying the required formulas, and, most importantly, formatting the resulting data to transform raw numerical outputs into easily interpretable dates. Adopting this approach will fundamentally enhance the precision and efficiency of your date-based computations in [Google Sheets](#).

## The Technical Foundation: Date Storage and Serial Numbers

Working with [dates](#) in any [spreadsheet](#) application introduces unique technical considerations related to how these values are internally managed. Dates are not stored as recognizable text strings; instead, they are converted and processed as sequential [numeric values](#), often referred to as serial dates. This serial number represents the total count of days that have elapsed since a defined reference point, or epoch date--typically January 1, 1900, in [Google Sheets](#) and Excel. While this numerical system is highly effective for mathematical operations (such as calculating durations), it means that the direct output of any date-related [formula](#) will appear as an unformatted number, requiring an essential formatting step later on.

Compounding this technical storage method is the global inconsistency regarding the definition of a "week." As noted, differing regional standards dictate which [day of the week](#) begins the cycle. For instance, the [ISO 8601 standard](#), widely adopted in Europe and various professional contexts, insists on Monday as the first day. Conversely, traditional standards prevalent in the U.S. and Canada recognize **Sunday** as the start. This variability necessitates the use of adaptable [formulas](#) capable of dynamically shifting their calculation based on the required starting day, thereby avoiding manual adjustments for diverse datasets.

To efficiently overcome these challenges, a thorough grasp of fundamental numerical functions, such as the [MOD function](#), and its application within the context of the [serial date](#) system is vital. Although the built-in [WEEKDAY function](#) can return a number corresponding to the day of the week (e.g., 1 for Sunday, 2 for Monday), the specialized [formulas](#) detailed here offer a more direct and streamlined approach specifically engineered to return the actual date of the week's beginning. By applying these robust mathematical expressions, we ensure unwavering accuracy across any dataset, simplifying the entire date management workflow.

## Formula 1: Calculating the Sunday-First Week Start

For users operating under the convention where **Sunday** is recognized as the first [day of the week](#) (standard in many countries, including the US), the following formula provides the precise date of that week's starting point:

**=A2-MOD(A2-1, 7)**

To understand the underlying mechanism of this solution, we must examine how the [MOD function](#) is applied to the [serial date](#) value. The key lies in the adjustment: subtracting `1` from the date in cell **A2**. Since [spreadsheet](#) systems internally assign numerical values to days, this adjustment shifts the internal day-of-week indexing such that Sunday now produces a remainder of 0 when divided by 7. The function `MOD(A2-1, 7)` thus returns the exact numerical offset--the number of days elapsed since the most recent Sunday.

For example, if the calculation results in a remainder of `0`, the date in **A2** is confirmed as a Sunday. A remainder of `1` indicates Monday, and so on. By subtracting this remainder from the original date in **A2**, the [formula](#) effectively "rolls back" the date to the most recent Sunday, thereby isolating the first day of that week's cycle. This method is highly efficient as it relies solely on mathematical precision, circumventing the need for complex nested IF statements or explicit reliance on the [WEEKDAY function](#).

## Formula 2: Calculating the Monday-First Week Start

For organizations and professionals who adhere to the convention where **Monday** is designated as the first [day of the week](#), typically aligning with global [ISO 8601 standards](#), a slight but critical modification to the formula is required. This adjustment ensures that the calculation correctly references Monday as the start of the seven-day period:

**=A2-MOD(A2-2, 7)**

The fundamental structure of this calculation mirrors the Sunday-first version, but the pre-modulo adjustment changes from subtracting `1` to subtracting `2` from the [date](#) in cell **A2**. This modification is essential for recalibrating the modulo operation's reference point. When Monday is the defined start day, subtracting `2` from the date's underlying [serial number](#) ensures that Monday consistently results in a remainder of 0 when the result is divided by 7 via the [MOD function](#).

If the date's adjusted serial number yields a remainder of `0` after being processed by `MOD(..., 7)`, it confirms that the original date (in **A2**) is a Monday. A remainder of `1` indicates Tuesday, and so forth. By subtracting this calculated remainder from the original date in **A2**, the final [formula](#) successfully returns the date of the most recent Monday. This design ensures both formulas are highly versatile, requiring only the input date (assumed to be in cell **A2**) to execute the calculation successfully.

## Practical Implementation: Setting Up Your Sheet and Applying Formulas

Successful implementation begins with properly structuring your [Google Sheets](#) environment. For optimal execution, your source dates must be clearly organized in an accessible column. Throughout our examples, we will assume that the raw input dates are located in column **A**, starting at cell **A2**. This standardized arrangement simplifies cell referencing and facilitates the efficient application of formulas across an entire range of data using autofill capabilities.

After preparing your input dates in column **A**, the next step involves designating clear output columns for the results. For maximum clarity and comparison, we recommend using column **C** to display the calculated first day of the week based on the Sunday-start convention, and column **D** for the results derived from the Monday-start convention. This disciplined layout ensures that your data remains organized and professional, allowing for immediate, side-by-side analysis of the outcomes generated by the two different weekly standards.

	A	B	C	D
1	<b>Date</b>	<b>Sales</b>		
2	1/4/2022	23		
3	2/15/2022	40		
4	3/23/2022	34		
5	4/15/2022	39		
6	4/18/2022	25		
7	5/1/2022	19		
8	8/7/2022	14		
9	10/31/2022	14		
10	11/4/2022	20		
11	12/19/2022	22		
12				
13				
14				
15				
16				
17				

The final step in this phase is the execution of the core formulas. Enter the Sunday-first formula directly into cell **C2** and the Monday-first formula into cell **D2**. These initial entries will establish the calculation for the date in **A2**:

**C2:** `=A2-MOD(A2-1,7)`

**D2:** `=A2-MOD(A2-2,7)`

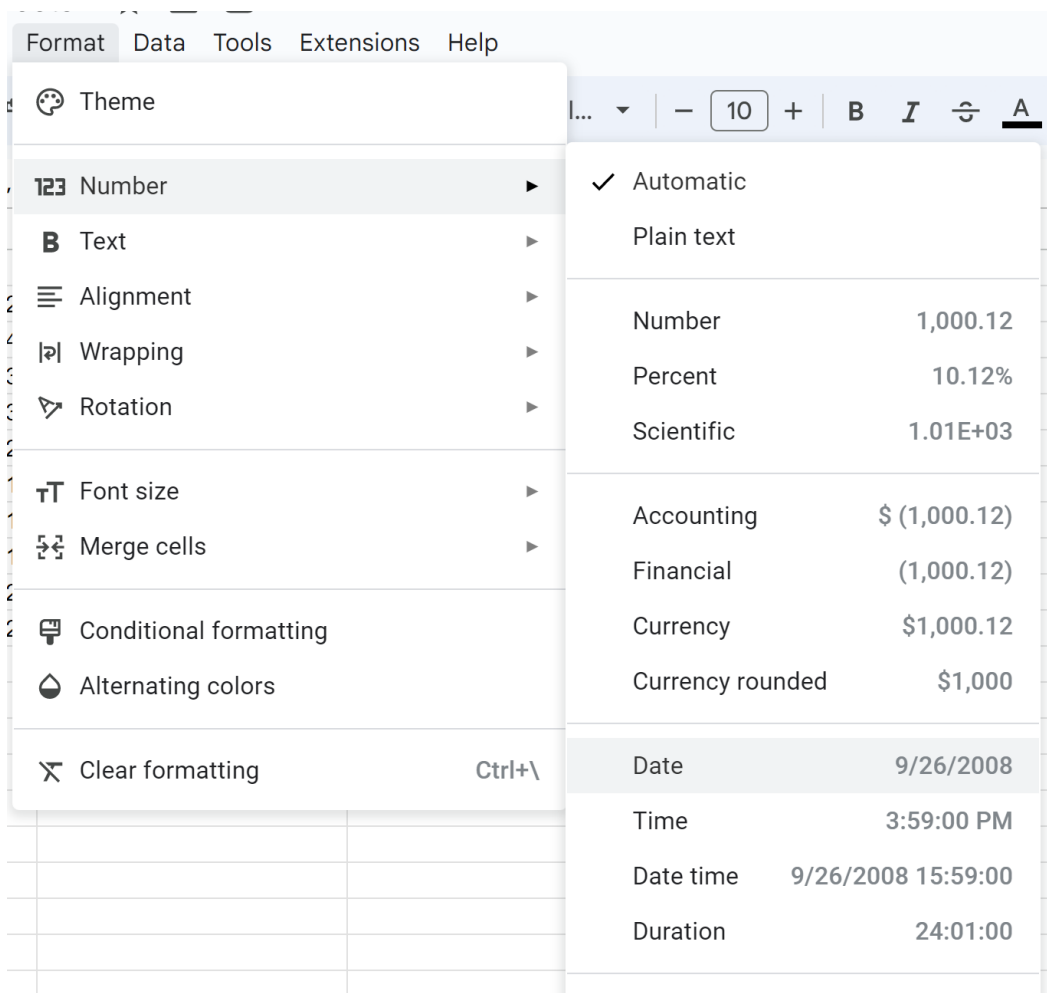
To apply these calculations to your entire dataset, utilize the powerful **fill handle** feature of [Google Sheets](#). Select cell **C2**, click the small square at the bottom-right corner, and drag it downward to cover all corresponding input dates in column **A**. Repeat this exact procedure for cell **D2**. This "autofill" function automatically adjusts the relative cell references, ensuring that the first day of the week is calculated accurately for every single date in your list without requiring tedious manual re-entry.

## Formatting Output: Converting Serial Numbers to Readable Dates

Upon the successful application and autofill of the date formulas, the calculated outcomes in columns **C** and **D** will initially appear as large, unformatted [numeric values](#). This is expected behavior, directly reflective of the [spreadsheet](#) application's internal reliance on the serial number system for managing chronological data. For instance, a number like `44562` is simply the numeric

representation of a calendar [date](#). This numerical foundation is critical because it enables the precise mathematical operations, such as the subtraction of days, upon which our formulas depend.

To convert these numeric outputs into a clear, universally recognizable date format, a specific sequence of formatting actions must be followed. First, accurately highlight the entire range of cells containing your numeric date values (e.g., **C2:D11**). Highlighting the entire output range guarantees that all calculated first days of the week are formatted consistently. Next, navigate to the [Google Sheets](#) menu bar and click the **Format** tab. From the resulting primary dropdown menu, select the **Number** option. A secondary submenu will appear, from which you must definitively choose the **Date** format. This action instructs [Google Sheets](#) to interpret the underlying numerical quantities as dates and display them using a standard, human-readable format.



Following the application of the **Date** formatting, your sheet will be visually transformed. Column **C** will clearly present the first day of the week based on the Sunday-start convention, while column **D** will show the corresponding result based on the Monday-start convention, all displayed in a

concise and professional date format. This crucial formatting step is essential for rendering your analytical results comprehensible and immediately actionable.

	A	B	C	D
1	<b>Date</b>	<b>Sales</b>	<b>First Day of Week (Sunday)</b>	<b>First Day of Week (Monday)</b>
2	1/4/2022	23	1/2/2022	1/3/2022
3	2/15/2022	40	2/13/2022	2/14/2022
4	3/23/2022	34	3/20/2022	3/21/2022
5	4/15/2022	39	4/10/2022	4/11/2022
6	4/18/2022	25	4/17/2022	4/18/2022
7	5/1/2022	19	5/1/2022	4/25/2022
8	8/7/2022	14	8/7/2022	8/1/2022
9	10/31/2022	14	10/30/2022	10/31/2022
10	11/4/2022	20	10/30/2022	10/31/2022
11	12/19/2022	22	12/18/2022	12/19/2022
12			<b>=A2-MOD(A2-1,7)</b>	<b>=A2-MOD(A2-2,7)</b>
13				
14				
15				
16				
17				

## Verifying Accuracy Through Calendar Cross-Reference

Once the formulas have been successfully executed and the outputs correctly formatted as readable [dates](#), a final verification step is strongly recommended to confirm the integrity and accuracy of the calculations. This involves cross-referencing a sample date from your dataset against a standard [calendar](#). For our example, let us verify the results for the input date **1/4/2022** (January 4, 2022).

By consulting a [calendar](#) for January 2022, we confirm that **1/4/2022** falls on a Tuesday. We must now determine if our two distinct formulas correctly identify the start of the week relative to this Tuesday, based on their programmed conventions. The outcome must logically adhere to the defined week start for the result to be considered accurate and reliable.



January 2022						
Su	Mo	Tu	We	Th	Fr	Sa
26	27	28	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	29
30	31	1	2	3	4	5

If we apply the convention that **Sunday** is the first day of the week, the Sunday immediately preceding 1/4/2022 is **1/2/2022**. Consequently, the formula in Column C (Sunday-first) must yield this exact date. Conversely, if **Monday** is established as the first day, the Monday immediately preceding 1/4/2022 is **1/3/2022**. The formula utilized in Column D (Monday-first) is therefore expected to show this latter date. The calculated dates presented in our [Google Sheets](#) example precisely match these calendar-verified outcomes, unequivocally confirming the robust reliability and mathematical accuracy of the formulas detailed in this guide.

## Conclusion and Next Steps for Date Mastery

Gaining proficiency in accurately determining the first [day of the week](#) in [Google Sheets](#), regardless of whether the week begins on Sunday or Monday, is an essential skill for anyone managing chronological data. The elegantly simple yet powerful formulas, which intelligently leverage the [MOD function](#), provide a highly flexible and robust solution to a common [spreadsheet](#) challenge. By coupling an understanding of the underlying [numeric date values](#) with the application of proper date formatting, users can ensure their data is consistently precise, aligning perfectly with diverse regional requirements.

The technical methods outlined within this guide not only solve the immediate problem of locating the week start but also deepen your foundational understanding of general date manipulation techniques in [Google Sheets](#). This core knowledge can be expanded to tackle numerous other time-based tasks, such as calculating the last day of the week, determining week numbers, or

isolating specific weekdays across large datasets. The versatility inherent in these mathematical formulas provides the tools necessary to execute complex data analysis with improved accuracy, effectively transforming raw chronological data into high-value, actionable business insights.

We strongly encourage readers to integrate these efficient formulas into their existing [Google Sheets](#) projects. The ability to dynamically switch between Sunday-start and Monday-start cycles is particularly beneficial for rigorous project management, detailed scheduling, and financial reporting where strict adherence to specific weekly conventions is required. As your proficiency develops, consider exploring other advanced date and time functions available, such as [WEEKDAY](#), `EOMONTH`, or `WORKDAY`, to further maximize your data processing capabilities.

## **Additional Resources**

The following supplementary tutorials explain how to perform other essential and common tasks within [Google Sheets](#):