

# How to Get the First Day of the Previous Month in Excel

Authored by  
**stats writer**

January 24, 2026

## RECOMMENDED CITATION

stats writer (2026). *How to Get the First Day of the Previous Month in Excel*.  
PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=127419>

## Excel: Get First Day of Previous Month

Mastering dynamic date calculation is fundamental when working with [Microsoft Excel](#) for financial reporting, project management, and time-series analysis. A common requirement in business intelligence is the ability to automatically identify the first day of the preceding month relative to a specific date or the current system date. This calculation provides a crucial starting point for calculating monthly performance metrics or historical comparisons.

### The Essential Formula for Previous Month's Start Date

To effectively retrieve the first day of the previous month, regardless of whether the reference is a static date or the current date, we utilize a powerful combination of date functions. The core logic hinges on the [EOMONTH function](#), which efficiently handles the complexities of varying month lengths.

You can use the following concise and robust formula to determine the first day of the month preceding the date provided in a specific cell reference:

```
=EOMONTH(A2,-2)+1
```

This particular formula, referencing cell **A2**, is designed to first navigate two months backward from the date specified in **A2**, find the end of that earlier month, and then increment the result by one day to land precisely on the first day of the previous month. This highly effective technique simplifies complex temporal comparisons and is a staple in advanced [Excel formulas](#).

For instance, consider a scenario where cell **A2** holds the date **February 12, 2024 (2/12/2024)**. Applying the formula above returns **January 1, 2024 (1/1/2024)**. The calculation dynamically adjusts for leap years and different month lengths, ensuring accuracy across all calendar scenarios, which is vital for reliable [data tracking](#) and reporting.

### Practical Demonstration: Setting up the Dataset

To illustrate the practical application of this dynamic date calculation, let us work with a simulated sales dataset. This dataset often represents typical business scenarios where transactional dates must be analyzed against specific reporting periods. Establishing a clear data structure is the first step toward successful implementation of any date-related formula in your spreadsheet.

Suppose we have the following sample data in [Microsoft Excel](#), containing records of sales transactions completed on various dates throughout a reporting quarter. Our objective is to generate a corresponding column that identifies the beginning of the previous reporting month for

each entry.

	A	B	C	D	E
1	<b>Date</b>	<b>Sales</b>			
2	1/5/2024	10			
3	1/18/2024	14			
4	2/12/2024	19			
5	3/29/2024	25			
6	4/25/2024	23			
7	5/12/2024	28			
8	5/15/2024	22			
9	6/1/2024	40			
10	7/14/2024	23			
11	8/22/2024	15			
12					
13					
14					
15					

In this setup, Column A contains the primary Date field. We aim to populate Column C, titled "First Day of Previous Month," using our specialized [Excel formula](#). This setup allows us to easily relate each transaction date back to the start of its antecedent month, which is frequently necessary when aggregating monthly sales summaries or determining eligibility windows.

## Step-by-Step Implementation of the Formula

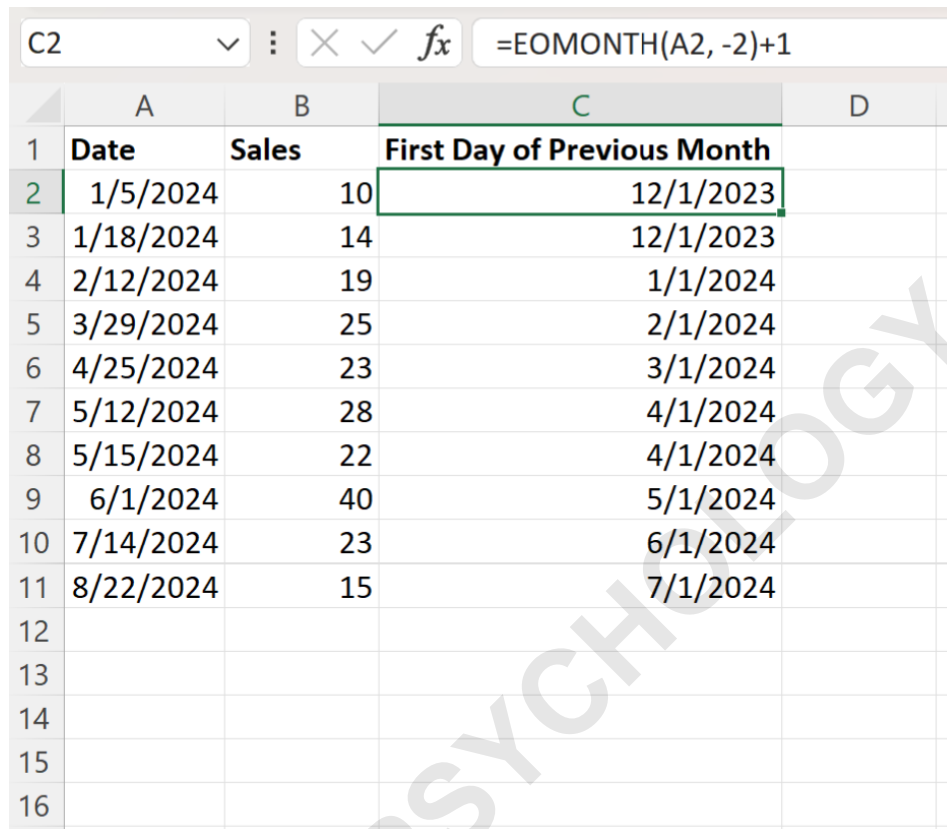
To find the first day of the previous month for every date listed in the **Date** column (Column A), we must apply the formula starting at the first row of our result set, typically cell C2. This method allows for efficient propagation across the entire dataset, maintaining consistency and accuracy.

The initial step involves typing or pasting the precise formula into cell **C2**, ensuring that it correctly references the corresponding date in cell A2. The structure remains identical to our core solution:

**=EOMONTH(A2,-2)+1**

Once entered, press Enter to calculate the result for the first date. This provides the foundation for the rest of the column. The output in C2 should show the first day of the month preceding the date in A2. This single point calculation confirms the logical structure of the [Excel formula](#) before applying it universally.

The subsequent crucial step involves utilizing Excel's powerful autofill feature. By clicking on the small square (fill handle) located at the bottom right corner of cell C2 and dragging it downwards, we efficiently copy the formula to every remaining cell in Column C. This action automatically adjusts the cell reference (A2 becomes A3, A4, and so on) for each subsequent row, ensuring every date calculation is relative to its specific row entry.



The screenshot shows an Excel spreadsheet with the following data:

	A	B	C	D
1	<b>Date</b>	<b>Sales</b>	<b>First Day of Previous Month</b>	
2	1/5/2024	10	12/1/2023	
3	1/18/2024	14	12/1/2023	
4	2/12/2024	19	1/1/2024	
5	3/29/2024	25	2/1/2024	
6	4/25/2024	23	3/1/2024	
7	5/12/2024	28	4/1/2024	
8	5/15/2024	22	4/1/2024	
9	6/1/2024	40	5/1/2024	
10	7/14/2024	23	6/1/2024	
11	8/22/2024	15	7/1/2024	
12				
13				
14				
15				
16				

After this operation, Column C is completely populated, showing the calculated first day of the previous month for each corresponding date in Column A. This automated process drastically reduces manual input errors and ensures the consistency required for accurate financial modeling and data tracking initiatives.

## Interpreting the Results and Date Logic

The newly populated Column C provides essential context for analyzing the transactional dates. Understanding how the dates shift illustrates the formula's effectiveness in handling different monthly boundaries.

The results confirm the intended operation of the formula:

For a transaction occurring on **January 5, 2024 (1/5/2024)**, the previous month's first day is

correctly identified as **December 1, 2023 (12/1/2023)**. Note that regardless of the day within January (e.g., the 5th or the 18th), the previous month's boundary remains fixed.

Similarly, for the date **January 18, 2024 (1/18/2024)**, the formula returns **December 1, 2023 (12/1/2023)**. This consistency is paramount for grouping data accurately by reporting month.

When moving into the next calendar month, such as a date of **February 12, 2024 (2/12/2024)**, the result dynamically shifts to **January 1, 2024 (1/1/2024)**.

This automated categorization ensures that all data points within a given month are referenced back to the same starting point for the preceding period, providing a standardized anchor for calculating variances, trends, or cumulative figures based on the prior month's performance data. This is an indispensable tool for business analysts focused on timely reporting.

## Deconstructing the EOMONTH Function

To fully appreciate the efficiency of this solution, we must examine the mechanism of the EOMONTH function, which forms the logical backbone of our calculation. Recall the formula used to find the first day of the previous month for cell **A2**:

**=EOMONTH(A2,-2)+1**

The **EOMONTH** function in Microsoft Excel is specifically designed to calculate the last day of a month that is a specified number of months before or after a given start date. It accepts two arguments: the **start\_date** (in our case, cell A2) and **months** (the integer representing the offset). A positive number for **months** moves forward, and a negative number moves backward.

The critical component here is **EOMONTH(A2, -2)**. By using a value of **-2** for the months offset, we instruct Excel to step back two full calendar months from the month containing the date in A2, and then return the last day of that resulting month. For example, if the start date in cell **A2** is **January 5, 2024 (1/5/2024)**: stepping back one month lands us in December 2023, and stepping back two months lands us in November 2023. Thus, **EOMONTH(A2, -2)** returns the last day of November 2023, which is **November 30, 2023 (11/30/2023)**.

The final step in the formula, adding **+1**, converts the last day of that earlier month into the first day of the subsequent month--which is precisely the beginning of the previous month relative to the original start date. Continuing the example, adding 1 to November 30, 2023, results in **December 1, 2023 (12/1/2023)**. This systematic approach ensures that the calculation is robust and independent of the day-of-the-month value in the original cell A2.

## Handling Dynamic Dates: Using TODAY() and NOW()

While referencing a fixed cell like A2 is useful for static datasets, many business applications

require calculations relative to the current moment. Microsoft Excel provides functions to handle these dynamic scenarios, primarily TODAY() and NOW().

If the goal is always to find the first day of the month preceding the system's current date, the formula is adapted by replacing the cell reference (A2) with the **TODAY()** function. The resulting Excel formula becomes `=EOMONTH(TODAY(), -2)+1`. This ensures that the report refreshes daily, automatically adjusting the reference month without manual intervention. This dynamic functionality is critical for dashboards and automated reporting systems where timeliness is paramount.

It is important to understand the subtle difference between **TODAY()**, which returns the current date without the time component, and **NOW()**, which returns both the current date and time. For date calculations focused solely on month boundaries, **TODAY()** is usually preferred as it maintains simplicity and avoids potential time-related ambiguities. Using **TODAY()** ensures that the logic of the **EOMONTH** function remains focused exclusively on the calendar month structure.

## Alternative Methods for Date Calculation

Although the combination of EOMONTH function and arithmetic addition is the cleanest method, especially for cross-version compatibility, other Excel formulas can achieve similar results, though often requiring more complex nested logic. One common alternative involves utilizing the **DATE**, **YEAR**, and **MONTH** functions.

An alternative approach to find the first day of the current month is `=DATE(YEAR(A2), MONTH(A2), 1)`. To adapt this to the previous month, we must subtract one from the month number, handling the year rollover if the current month is January. This leads to a more complex formula structure, such as `=DATE(YEAR(A2), MONTH(A2)-1, 1)`. While this works for months February through December, it fails in January because `MONTH(A2)-1` would result in 0, which is invalid.

Therefore, the robust, year-rollover-proof alternative is significantly longer: `=DATE(YEAR(A2), MONTH(A2)-1, 1)`. If A2 is January, we need to manually adjust the year and month: `=DATE(YEAR(A2)-(MONTH(A2)=1), IF(MONTH(A2)=1, 12, MONTH(A2)-1), 1)`. Comparing this lengthy expression to the elegant `=EOMONTH(A2, -2)+1` clearly demonstrates the superiority and readability of the **EOMONTH** method for this specific date manipulation task.

## Common Use Cases and Applications

The ability to dynamically calculate the first day of the previous month is not just a theoretical exercise; it has substantial practical applications across various professional fields, particularly those involving financial analysis and data tracking.

**Financial Reporting:** Calculating month-over-month (MoM) growth requires comparing the current

period's data with the previous period's data. By identifying the previous month's start date, analysts can use functions like **SUMIFS** or **COUNTIFS** to accurately aggregate data that falls within the required preceding window (e.g., all sales between `=EOMONTH(TODAY(), -2) + 1` and `=EOMONTH(TODAY(), -1)`).

**Inventory and Logistics:** Businesses often track inventory levels or shipping volumes based on the beginning of the reporting cycle. Automatically determining the previous month's start date aids in analyzing stock turnover rates or identifying bottlenecks that occurred during the immediate prior reporting period. This improves predictive modeling and resource allocation strategies.

**Automated Data Filtering:** In larger spreadsheets or linked data sources, this formula can serve as a criteria generator. For example, in pivot tables or advanced filter settings, the calculated date can filter records to show only transactions relevant to the past 30 or 31 days, relative to the first day of the previous month, providing a consistent analysis window.

The adaptability of the `=EOMONTH(A2, -2) + 1` construction allows it to be nested within complex dashboards, ensuring that critical reports always reference the correct historical anchor points, thereby drastically improving the efficiency and reliability of recurring monthly analysis.

## Summary and Key Takeaways

The formula `=EOMONTH(A2, -2) + 1` represents the most efficient and robust solution in Microsoft Excel for programmatically finding the first day of the previous calendar month relative to a specific start date. Its reliance on the powerful EOMONTH function elegantly bypasses the need for complex conditional logic related to year rollovers or varying month lengths.

By first stepping back two months using the **-2** offset parameter within **EOMONTH**, we locate the end date of the month before the previous month. The subsequent addition of **+1** then perfectly increments that date to the first day of the desired preceding month. This technique is invaluable for anyone performing advanced temporal analysis, generating consistent reports, or automating data tracking processes within a spreadsheet environment.

**Note:** You can find the complete documentation for the EOMONTH function in Excel via the official Microsoft Support website for detailed parameter guidance and troubleshooting information.

## Further Resources and Tutorials

The following tutorials explain how to perform other common operations in Excel related to date manipulation and advanced analysis techniques: