

How to Find the First Day of the Week in Power BI

Authored by
mohammed loot

January 10, 2026

RECOMMENDED CITATION

mohammed loot (2026). *How to Find the First Day of the Week in Power BI*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=125459>

Determining the first day of the week in Power BI is a foundational task for implementing robust time-intelligence calculations. This process involves calculating the exact date of the start of the week for any given date within your dataset. This capability is absolutely crucial for generating accurate weekly reports, performing rolling aggregations, and ensuring data analysis is consistent across various regional calendar standards.

While standard functions like DATEADD or DATEDIFF might be useful for general date manipulation, the most reliable approach for weekly alignment involves leveraging the WEEKDAY function within DAX. The goal is simple: take any date (e.g., Wednesday, July 17th) and derive the date of that week's designated start (e.g., Sunday, July 14th, or Monday, July 15th). Achieving this requires carefully defining which day is considered the start of the week based on your company's requirements or regional conventions.

Understanding the Need for Weekly Start Dates in Power BI

Analyzing transactional data on a weekly basis provides vital insights into short-term performance trends and cyclical business patterns. However, discrepancies arise quickly if the definition of the "start of the week" is inconsistent. In the United States and Canada, the week traditionally begins on Sunday, whereas many European and international standards (following ISO 8601) dictate that the week begins on Monday. For organizational reporting, it is paramount that all weekly aggregations align with a single, defined starting point.

When working with large datasets in Power BI, we often need to group measures (like total sales or customer counts) by the week they belong to. Without a specific column identifying the week's starting date, grouping becomes ambiguous. By creating a calculated column that reliably extracts the first day of the week, we enable powerful filtering, drilling, and visualization capabilities, ensuring that every weekly bucket in our reports is accurately defined and comparable across time periods. This calculated column acts as a consistent primary key for weekly analysis.

The DAX language offers specialized functions that handle this date arithmetic efficiently. We calculate the offset (the number of days elapsed since the start of the current week) and subtract that offset from the original date. This mathematical approach guarantees accuracy regardless of the year or the presence of leap years, providing a robust solution superior to complex conditional logic often found in other programming environments.

Introduction to DAX Date Functions for Week Calculations

The core of calculating the first day of the week in Power BI lies in mastering the WEEKDAY function. This function returns a numerical value corresponding to the day of the week for a given date. Its power comes from the optional second argument, the return_type, which dictates which day the numbering begins with (i.e., whether Sunday is 1 or Monday is 1).

The syntax is defined as: `WEEKDAY(<date>,)`. The return type is crucial because it directly influences how we structure the final calculation. For example, if we use return type 1 (default, Sunday=1), subtracting the weekday number will correctly align the date to the preceding Saturday, necessitating an adjustment to land on Sunday. Conversely, return type 2 (Monday=1) is often preferred for calculating Monday starts, as it simplifies the subsequent arithmetic.

To calculate the start of the week, we perform a simple date subtraction: `Start Date = Date - (Day Number Offset)`. The challenge is ensuring the offset accurately reflects the number of days we need to step backward to reach the desired start day. Because DAX treats dates as numerical values (where 1 unit equals one day), simple subtraction works flawlessly to shift the date backward in time. The following formulas demonstrate the precise configuration needed to handle the two most common organizational requirements: a Sunday start and a Monday start.

Formula 1: Defining the Week Start (Sunday Default)

For reporting environments where the week legally or conventionally starts on Sunday, we must select a WEEKDAY function return type that facilitates this structure. The most straightforward method uses return type 2, where Monday is 1 and Sunday is 7. Although this return type is typically associated with Monday-based weeks, we can leverage it effectively to backtrack to Sunday.

Using return type 2 (Monday=1, Sunday=7), if today is Wednesday (which returns 3), we need to subtract 3 days to get back to Sunday (since Sunday is day 7 of the previous week). However, if today is Monday (which returns 1), we need to subtract 1 day to land on the previous Sunday. This necessitates a slight adjustment or selection of a specific return type. The structure below utilizes return type 2, which assigns 1 to Monday and 7 to Sunday, but requires careful interpretation in the context of subtraction:

You can use the following formula in DAX to get the first day of the week for a given date, assuming the first day is Sunday:

Formula 1: Get First Day of Week (Assuming First Day is Sunday)

Week Start = 'my_data' - WEEKDAY('my_data', 2)

This formula creates a new column named **Week Start**. When using return type 2, the subtraction of the weekday number effectively pulls the date back to the Saturday preceding the current date's week. For instance, if the date is Wednesday, 1/10/2024, the WEEKDAY function returns 3. Subtracting 3 days gives us 1/7/2024, which is the previous Sunday. This specific arrangement of the formula works reliably for Sunday-start reporting when combined with return type 2 (Monday=1) in Power BI environments.

Formula 2: Adjusting the Week Start to Monday (ISO 8601 Compatible)

For organizations following global standards or operating primarily in regions that adhere to ISO 8601, the week must start on Monday. This is often the preferred choice for multinational business reporting. Fortunately, the structure of the WEEKDAY function with return type 2 is inherently set up to support a Monday start, requiring only a small adjustment to the arithmetic established in the previous formula.

When using return type 2, if the date is a Monday, the function returns 1. If we subtract 1 day, we land on the previous Sunday. However, we want the resulting date to be the current Monday. Therefore, after performing the subtraction to land on the previous Sunday (as shown in Formula 1), we must add exactly one day back to shift the result forward to the desired Monday start date. This simple addition ensures perfect alignment with the Monday-start convention.

Formula 2: Get First Day of Week (Assuming First Day is Monday)

Week Start = 'my_data' - WEEKDAY('my_data', 2) + 1

Both formulas create a new column named **Week Start** that contains the first day of the week for the corresponding date in the **Date** column. This calculation is dynamic and will automatically adjust for every single record in your dataset, whether your source table is small or contains millions of rows.

Deep Dive into the WEEKDAY Function and Return Type Parameters

Understanding the optional `return_type` parameter of the WEEKDAY function is essential for achieving the correct calculation in DAX. Although we primarily used return type 2 in the previous examples, Power BI actually supports three standard return types that define how the days of the week are numerically mapped:

Return Type 1 (Default): Sunday=1, Saturday=7. This is typical in North American contexts.

Return Type 2: Monday=1, Sunday=7. This is the global standard, highly favored for business logic due to its alignment with the work week.

Return Type 3: Monday=0, Sunday=6. This zero-based system is sometimes utilized in environments requiring arrays or specific indexing, though less common for simple week-start calculations.

The choice of return type directly affects the offset calculation. For example, if you chose return type 1, the subtraction logic would need to be slightly modified--instead of simply subtracting the

weekday value, you might need to use `MOD(WEEKDAY(Date, 1) + 6, 7)` to properly find the offset back to Sunday, ensuring the Sunday date itself doesn't roll back an extra week. Using return type 2, as demonstrated, provides a highly efficient and readable method for both Sunday and Monday starts, minimizing the complexity of the subtraction.

Practical Implementation: Step-by-Step Guide in Power BI

Let us walk through a practical example demonstrating how to apply these DAX formulas within a Power BI data model. Suppose we are tracking sales data and need to aggregate it weekly based on a Sunday start date. This scenario requires adding a new calculated column to our existing fact table.

Example: How to Get First Day of Week in Power BI

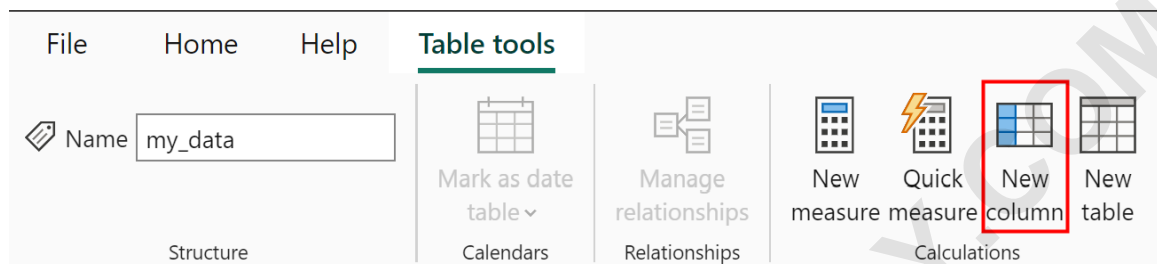
Suppose we have the following dataset in Power BI that contains information about total sales made on various dates by some company. Notice the structure includes a **Date** column and a **Total Sales** column:

The screenshot shows the Power BI interface with a table named 'my_data'. The ribbon is set to 'Table tools' and includes options for 'Mark as date table', 'Manage relationships', and 'Calculations'. The table data is as follows:

Date	Sales
Monday, January 8, 2024	10
Wednesday, January 10, 2024	14
Saturday, January 13, 2024	30
Monday, January 15, 2024	26
Friday, January 19, 2024	25
Tuesday, January 30, 2024	14
Sunday, February 4, 2024	10
Monday, February 5, 2024	11
Tuesday, February 13, 2024	15
Thursday, February 15, 2024	19
Monday, February 19, 2024	24
Sunday, February 25, 2024	12

Suppose that we would like to create a new column that contains the first day of the week for each corresponding date in the **Date** column of the table. This new column will serve as the dimension key for all weekly aggregations and comparisons.

To do so, navigate to the data view in Power BI Desktop, click the **Table tools** tab in the ribbon interface, and then click the **New column** icon. This action opens the formula bar, allowing you to define the DAX expression for the new calculated field:



To implement the Sunday start calculation, type the following formula into the formula bar. We assume the table is named `my_data` and the date column is `Date`:

Week Start = 'my_data' - WEEKDAY('my_data', 2)

This will create a new column named **Week Start** that contains the first day of the week (assuming the first day is considered **Sunday**) for each corresponding date in the **Date** column. Observe how all dates within the same seven-day window now map back to the same Sunday date:

1 Week Start = 'my_data'[Date] - WEEKDAY('my_data'[Date], 2)

Date	Sales	Week Start
Monday, January 8, 2024	10	1/7/2024 12:00:00 AM
Wednesday, January 10, 2024	14	1/7/2024 12:00:00 AM
Saturday, January 13, 2024	30	1/7/2024 12:00:00 AM
Monday, January 15, 2024	26	1/14/2024 12:00:00 AM
Friday, January 19, 2024	25	1/14/2024 12:00:00 AM
Tuesday, January 30, 2024	14	1/28/2024 12:00:00 AM
Sunday, February 4, 2024	10	1/28/2024 12:00:00 AM
Monday, February 5, 2024	11	2/4/2024 12:00:00 AM
Tuesday, February 13, 2024	15	2/11/2024 12:00:00 AM
Thursday, February 15, 2024	19	2/11/2024 12:00:00 AM
Monday, February 19, 2024	24	2/18/2024 12:00:00 AM
Sunday, February 25, 2024	12	2/18/2024 12:00:00 AM

Analyzing the Results and Common Pitfalls

The results shown in the newly created **Week Start** column are consistent and demonstrate the correct application of the Sunday-start formula. Let's analyze a few specific dates to ensure the logic holds:

The first day of the week for the date 1/8/2024 (a Monday) is **1/7/2024** (the preceding Sunday).

The first day of the week for the date 1/10/2024 (a Wednesday) is **1/7/2024** (the preceding Sunday).

The first day of the week for the date 1/13/2024 (a Saturday) is **1/7/2024** (the preceding Sunday).

The first day of the week for the date 1/15/2024 (the next Monday) is **1/14/2024** (the preceding Sunday).

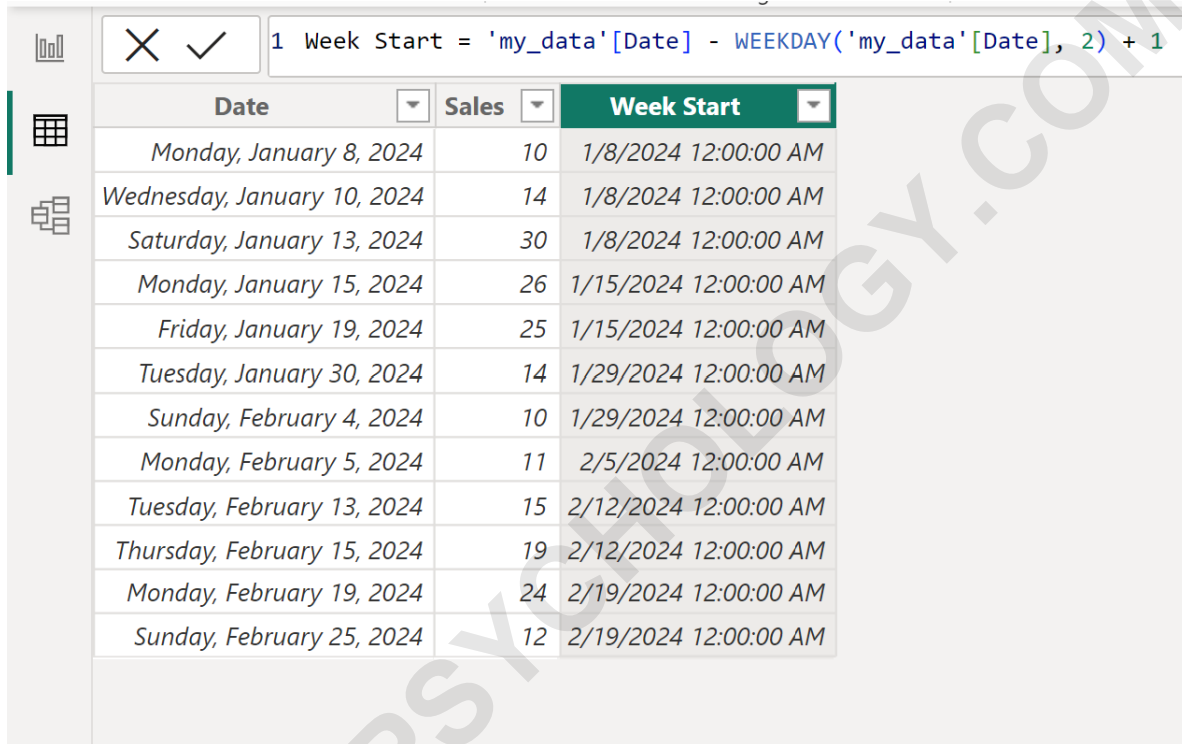
This shows the calculation is robust. However, a common pitfall is confusing the return types or forgetting to account for the necessary adjustment when switching between Sunday and Monday starts. If you had incorrectly used return type 1 (Sunday=1) with the simple subtraction method, dates falling on Sunday would register a Weekday value of 1, and subtracting 1 day would move the result back to the previous Saturday, leading to incorrect weekly grouping.

If your organizational standard requires a Monday start, the necessary modification is straightforward. If you would like the first day of the week to be considered Monday instead, then you can type the following formula into the formula bar. This small addition of + 1 ensures the

result lands precisely on Monday:

Week Start = 'my_data' - WEEKDAY('my_data', 2) + 1

This will create a new column named **Week Start** that contains the first day of the week (assuming the first day is considered **Monday**) for each corresponding date in the **Date** column. This outcome is highly desired for [ISO 8601](#) compliant analysis:



The screenshot shows the Power BI interface with a calculated column formula bar and a data table. The formula bar contains the formula: `1 Week Start = 'my_data' [Date] - WEEKDAY('my_data' [Date], 2) + 1`. The data table has three columns: Date, Sales, and Week Start. The Week Start column shows the first day of the week for each date, starting from Monday, January 8, 2024, and continuing through Sunday, February 25, 2024.

Date	Sales	Week Start
Monday, January 8, 2024	10	1/8/2024 12:00:00 AM
Wednesday, January 10, 2024	14	1/8/2024 12:00:00 AM
Saturday, January 13, 2024	30	1/8/2024 12:00:00 AM
Monday, January 15, 2024	26	1/15/2024 12:00:00 AM
Friday, January 19, 2024	25	1/15/2024 12:00:00 AM
Tuesday, January 30, 2024	14	1/29/2024 12:00:00 AM
Sunday, February 4, 2024	10	1/29/2024 12:00:00 AM
Monday, February 5, 2024	11	2/5/2024 12:00:00 AM
Tuesday, February 13, 2024	15	2/12/2024 12:00:00 AM
Thursday, February 15, 2024	19	2/12/2024 12:00:00 AM
Monday, February 19, 2024	24	2/19/2024 12:00:00 AM
Sunday, February 25, 2024	12	2/19/2024 12:00:00 AM

Advanced Considerations: Calendar Tables and ISO 8601

While calculated columns are useful for quick solutions, best practice in [Power BI](#) time intelligence often involves using a dedicated [DAX](#) Calendar Table. A proper Calendar Table contains every date relevant to your data model and pre-calculates all necessary time attributes, including the Start of Week, Week Number, Month, and Quarter. This approach improves model performance and simplifies measure creation significantly.

For advanced scenarios, especially those involving cross-border operations, strict adherence to the [ISO 8601](#) standard is necessary. ISO 8601 dictates that the week starts on Monday, and the first week of the year is the one containing the first Thursday of the year. While the formulas provided above handle the Monday start date correctly, determining the ISO Week Number requires a more complex [DAX](#) calculation, often utilizing the `WEEKNUM` function with the correct return type (21) or leveraging Power Query's built-in date functions for this specific calculation before loading the data

into the model.

By defining the Start of Week as a dedicated column, whether directly in the fact table or within a separate Calendar Table, analysts gain maximum flexibility. This allows them to use this column as the primary axis for visual reports, ensuring that weekly aggregates are always accurate and adhere to the established business logic (Sunday or Monday start). Always prioritize clean data structure over complex measure logic when dealing with date hierarchies.

Note: You can find the complete documentation for the [WEEKDAY function](#) in [DAX](#) on the official Microsoft documentation site.

Related Power BI Tutorials

The following tutorials explain how to perform other common date-related tasks in [Power BI](#), leveraging similar [DAX](#) and M-Query techniques:

How to use the [DATEADD function](#) to calculate previous year sales.

Using the [DATEDIFF function](#) to determine elapsed time between two dates.

Building a dynamic date table for advanced time-intelligence analysis.