

# How to Find the Maximum Value in Power BI: A Step-by-Step Guide

Authored by  
**stats writer**

January 28, 2026

## RECOMMENDED CITATION

stats writer (2026). *How to Find the Maximum Value in Power BI: A Step-by-Step Guide*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=128172>

Power BI is a sophisticated data visualization and business intelligence tool developed by Microsoft. It empowers users to transform raw data into meaningful, actionable insights, enabling superior analysis and interpretation. One fundamental capability essential for robust reporting is the ability to calculate and prominently display the maximum value within a specified data set. This feature is indispensable for quickly identifying the highest recorded metric--be it peak sales, maximum temperature, or the most points scored in a competition. To achieve this statistical calculation in Power BI, users primarily leverage the MAX function, which efficiently evaluates a column or expression and returns the single highest numerical value. For instance, if you are analyzing a detailed sales report that includes a column for total transaction value, the MAX function can instantaneously pinpoint the largest sale amount recorded. This capability significantly improves data comprehension, supports data-driven decision-making, and provides crucial visibility into peak performance trends within any organization.

## Calculate Max in Power BI (With Example)

### Introduction to Power BI and Calculating Maximums

Power BI serves as a critical bridge between complex data sources and understandable business metrics. When dealing with operational or financial data, identifying extreme values--specifically the maximum--is often the first step in performance review and anomaly detection. Knowing the maximum value allows analysts to establish benchmarks, understand limits, and evaluate outliers that may indicate exceptional performance or, conversely, data errors. The elegance of Power BI lies in its integration with the powerful formula language, DAX (Data Analysis Expressions), which facilitates these complex calculations with straightforward syntax.

The core necessity for calculating a maximum value spans numerous business applications. In sales, it helps identify the largest single transaction; in manufacturing, it might flag the highest production yield recorded in a shift; and in IT operations, it could highlight peak server load. Power BI measures are the tools we use to define these calculations, ensuring that the maximum value is dynamically updated whenever the underlying data model refreshes. This dynamic capability ensures reports are always current and reflective of the latest state of the business.

To effectively calculate the maximum, we must utilize the specific DAX function designed for this purpose: the MAX function. This function operates on a column within a table, scanning every row to locate and return the highest numerical or lexical value present. Understanding how to structure this formula is key to mastering data aggregation within the Power BI environment.

### Understanding the DAX Language for Data Analysis

DAX is the proprietary formula language used across Microsoft analytical tools, including Power BI,

Analysis Services, and Power Pivot in Excel. It is fundamentally different from languages like SQL or M (Power Query) because DAX is a functional language primarily focused on defining calculations, such as measures and calculated columns, within the context of a relational data model. When calculating metrics like the maximum value, we are defining a measure, which is a calculation that is executed at the time of filtering and visualization.

Measures are the dynamic heart of Power BI reports. Unlike calculated columns, which consume memory by storing results for every row in the table, measures are calculated on the fly, based on the filters applied by the user in the report canvas (e.g., filtered by date, region, or product category). This context sensitivity is paramount for accurate business intelligence reporting, ensuring that the maximum value displayed is always relevant to the current view.

The syntax for defining a measure to find the maximum value requires specifying the name of the new measure, the operator (in this case, equals), the DAX function, and the precise column reference. The column reference structure must always follow the format: 'TableName'. This explicit referencing ensures clarity and reduces ambiguity, especially in complex data models involving multiple related tables.

## The Power of the MAX Function in DAX

The MAX function is categorized as an aggregation function in DAX. Its primary purpose is straightforward: to find the largest value in a column. It is important to note that the MAX function only operates correctly on columns that contain numerical, date/time, or text values. When applied to text columns, it returns the value closest to the end of the alphabetical order. For standard quantitative analysis, however, it is predominantly used on numerical columns.

The core syntax required to implement this function and calculate the max value in a column is demonstrated below, creating a new measure in your Power BI data model:

**Max Points = MAX('my\_data')**

This specific formula creates a new, distinct measure named **Max Points**. This measure is defined to compute the maximum value found across all entries within the **Points** column, which belongs to the table designated as **my\_data**. This calculation is highly performant and immediately available for use in any visualization across your report pages, reflecting the current filter context applied by the report user.

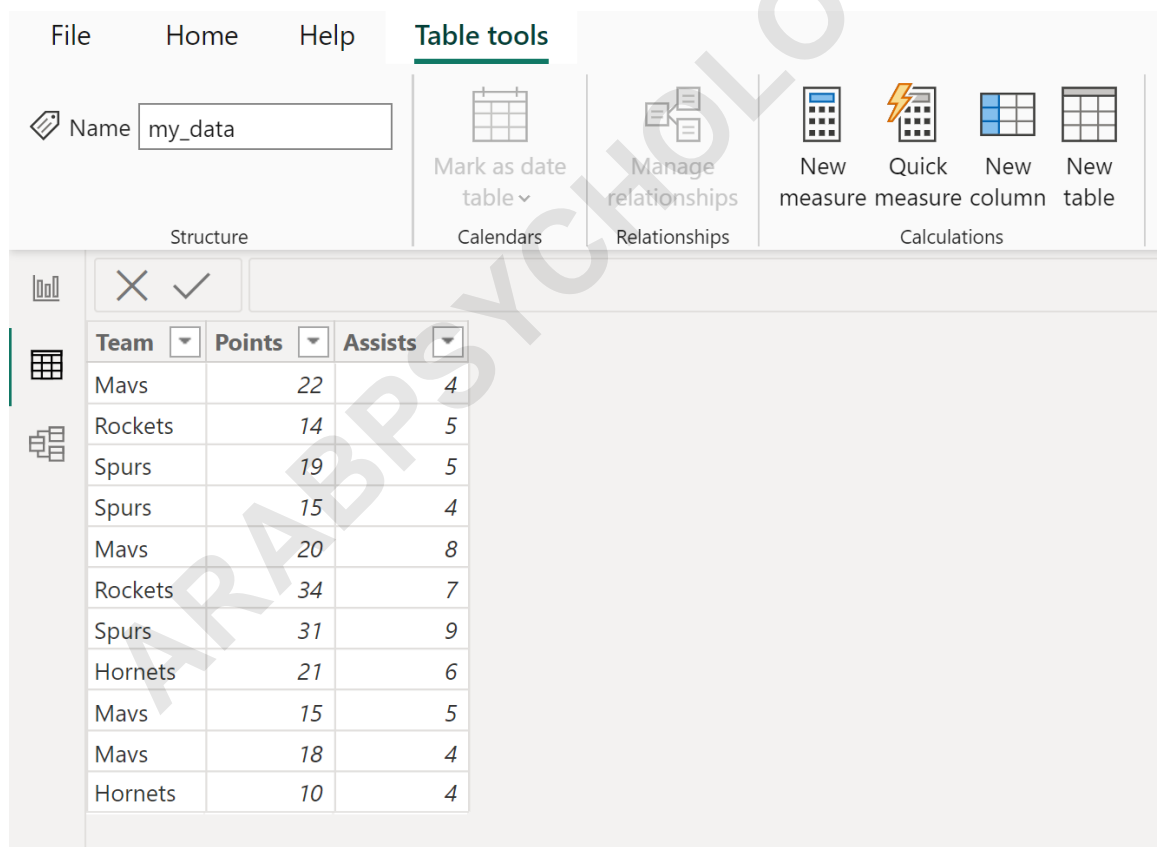
It is critical to distinguish MAX from its counterpart, MAXX. While MAX is restricted to column aggregation, MAXX is an iterator function that evaluates an expression row by row before determining the maximum result. For simple column maximums, MAX is the most efficient and appropriate choice, whereas MAXX is reserved for more complex scenarios, such as finding the

maximum based on a calculated ratio or a complex conditional expression.

## Practical Example Setup: Preparing the Data Model

To illustrate the application of the MAX function, let us utilize a practical scenario involving athlete performance data. Suppose we have imported a simple table into Power BI, which we have named **my\_data**. This table contains records related to basketball players across various teams, specifically tracking their performance points scored during games. Preparing the data model requires ensuring that the 'Points' column is correctly formatted as a numerical data type, typically a whole number or decimal number, to guarantee accurate calculation by the MAX function.

For the purpose of this demonstration, the **my\_data** table includes columns such as 'Team', 'Player Name', and 'Points'. Identifying the highest score recorded across all players is a common analytical requirement in data visualization. The raw data appears structured as follows, containing the records we intend to analyze:



The screenshot displays the Power BI interface. The 'Table tools' ribbon is active, showing options like 'Mark as date table', 'Manage relationships', 'New measure', 'Quick measure', 'New column', and 'New table'. Below the ribbon, a data table is visible with the following data:

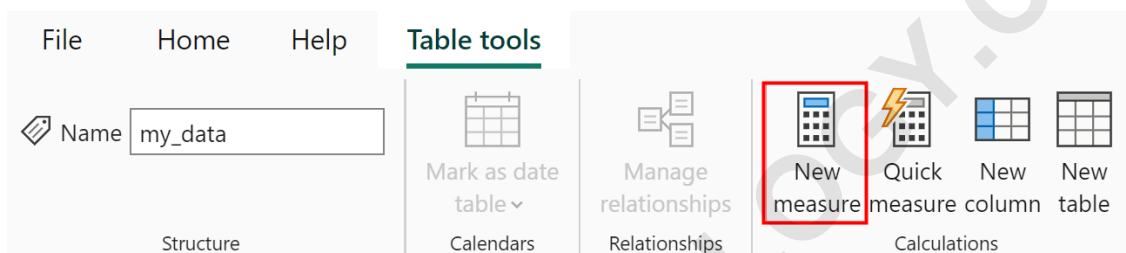
Team	Points	Assists
Mavs	22	4
Rockets	14	5
Spurs	19	5
Spurs	15	4
Mavs	20	8
Rockets	34	7
Spurs	31	9
Hornets	27	6
Mavs	15	5
Mavs	18	4
Hornets	10	4

Our objective is clear: we must calculate the single highest score present within the **Points** column. This value represents the highest recorded individual performance in the dataset, a key metric for performance benchmarking within this fictional sports analysis scenario. The subsequent steps will guide us through creating the necessary measure to extract this vital information.

## Step-by-Step Guide: Creating a New Measure using MAX

Calculating the maximum value begins within the **Power BI** Desktop interface. Since we are creating a dynamic calculation that is not tied to a specific row, we must define it as a measure. The process starts by navigating to the correct tab and initiating the measure creation command.

First, ensure you are viewing the Data or Report tab in Power BI Desktop. Locate and click on the **Table tools** tab positioned along the top ribbon menu. Within this ribbon, you will find the **New measure** icon. Clicking this icon opens the formula bar, allowing you to input your specific **DAX** expression. This action formally initiates the process of adding a new calculated metric to your data model.

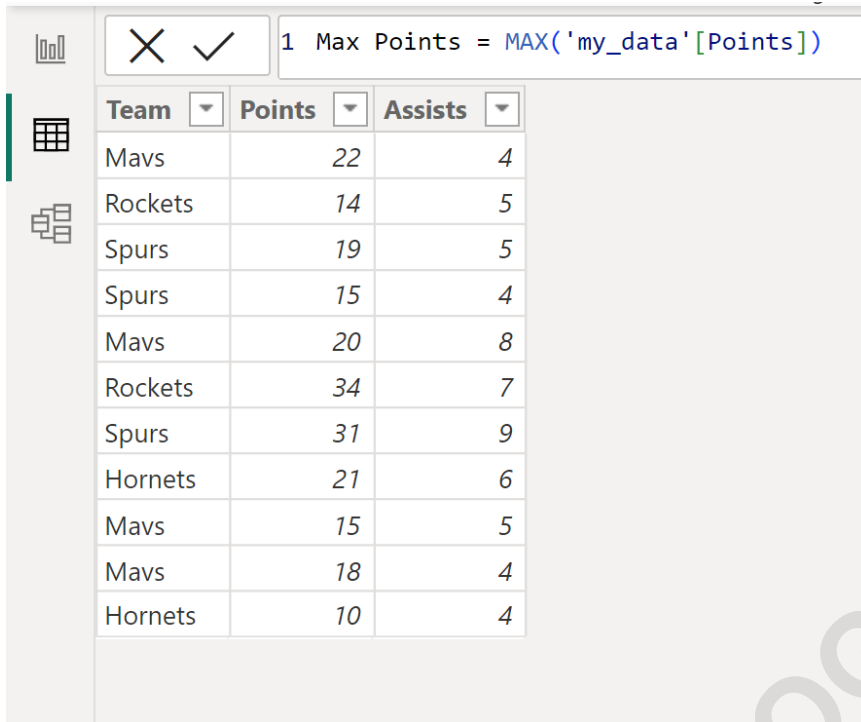


Once the formula bar is active, the next step involves typing the precise DAX formula. We will assign the name of the new measure and then use the **MAX function**, ensuring that the column reference correctly points to the 'Points' column within the 'my\_data' table:

**Max Points = MAX('my\_data')**

Upon committing this formula, the calculation is instantly added to your data model as a measure. This newly created measure, named **Max Points**, now permanently contains the computed highest value found in the **Points** column of the specified table. The result is immediately available for visualization, providing a crucial summary statistic derived directly from the underlying data.

The result of this definition is demonstrated in the Fields pane below, where the **Max Points** measure is displayed alongside the table columns, ready for deployment in reports:



The screenshot shows the Power BI interface. At the top, a DAX measure is defined: `1 Max Points = MAX('my_data'[Points])`. Below this, a table is displayed with the following data:

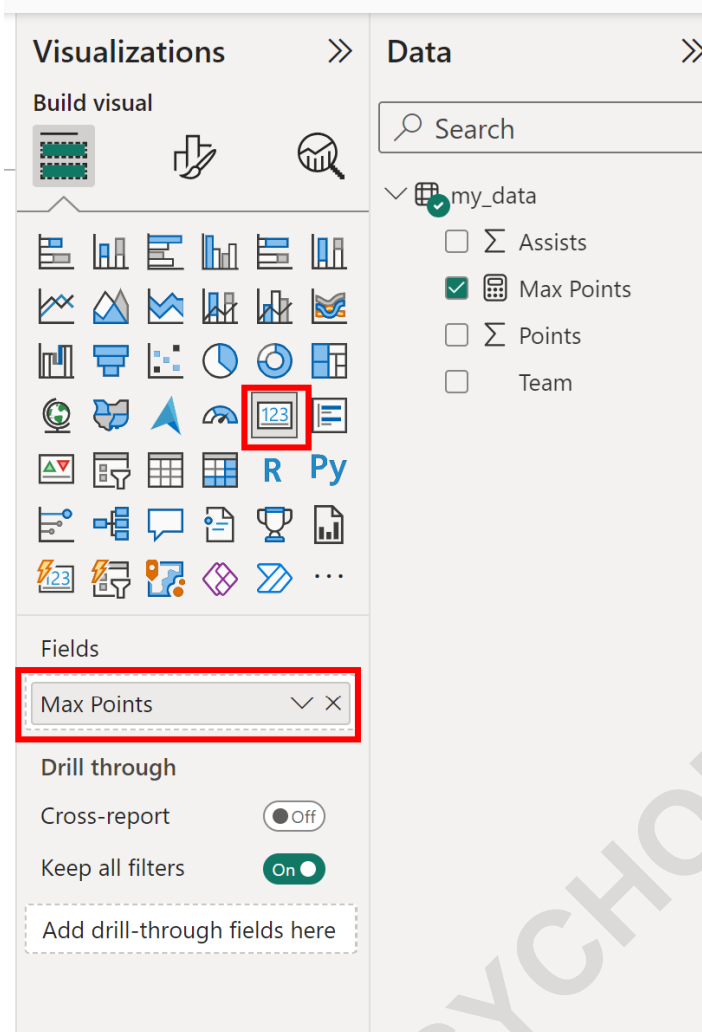
Team	Points	Assists
Mavs	22	4
Rockets	14	5
Spurs	19	5
Spurs	15	4
Mavs	20	8
Rockets	34	7
Spurs	31	9
Hornets	21	6
Mavs	15	5
Mavs	18	4
Hornets	10	4

## Analyzing the Result: Displaying the Maximum Value using a Card Visual

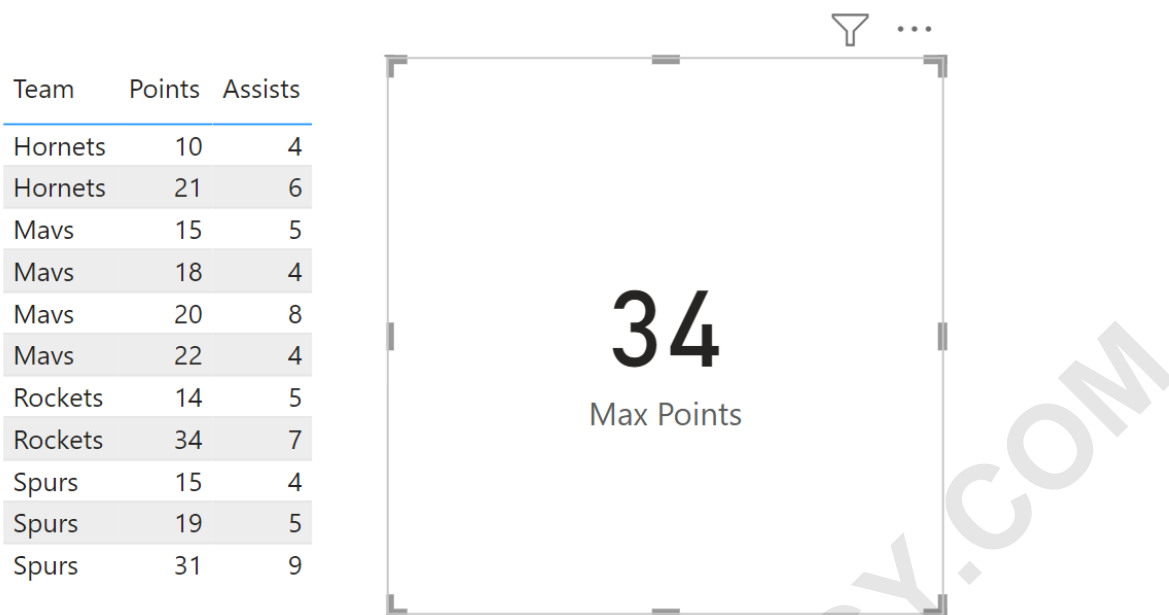
After successfully creating the **Max Points** measure using DAX, the final step is to present this result clearly within the data visualization layer of Power BI. While the measure exists in the model, it needs a visual element to communicate its value to the report consumer. The Card visual is arguably the most effective way to display a single, summarized metric like a maximum value.

To visualize the result, navigate to the **Report View** within Power BI Desktop. In the right-hand panel, under the **Visualizations** tab, click on the **Card** icon. This action places a placeholder Card visual onto your report canvas. This visual is specifically designed to highlight single numerical results with high impact and clarity.

Once the Card is selected, you must connect the visual to the calculated measure. This is achieved by dragging the newly created **Max Points** measure from the Fields pane and dropping it onto the **Fields** well (or area) of the selected Card visualization. This linkage instructs the Card to execute the underlying DAX calculation and display its output.



Upon configuring the Card visual, the report canvas will now prominently feature the calculated maximum value. This provides immediate confirmation of the highest recorded data point, derived dynamically from the source table.



As clearly displayed by the Card visual, the maximum value identified in the **Points** column across the entire **my\_data** table is **34**. This result is crucial for understanding the peak performance attained within the dataset and serves as a vital component of any comprehensive performance report in business intelligence.

## Advanced Applications and Alternatives to MAX

While the basic MAX function is excellent for column-level aggregation, advanced analytical scenarios often demand functions that can iterate over rows before aggregating. For instance, if you needed to find the maximum sales margin (calculated as `Sales - Cost`), you could not use MAX directly on the column. Instead, you would use the MAXX function.

The MAXX function allows you to specify a table and an expression to evaluate for each row of that table. This provides necessary flexibility when dealing with complex calculations, context transitions, or virtual tables. The syntax for MAXX is `MAXX(Table, Expression)`. For example, to find the maximum possible profit, the formula might look like: `Max Profit = MAXX('Sales', 'Sales' - 'Sales')`.

Another key consideration in advanced Power BI development is understanding evaluation context. Measures created with MAX are highly susceptible to filter context. If you use the **Max Points** measure in a matrix visual filtered by 'Team', the measure will automatically recalculate and display the maximum points scored for *each* respective team, rather than the overall maximum of 34. This contextual evaluation is a powerful feature of DAX that analysts must master to build accurate

and detailed reports.

## Conclusion: Leveraging Maximum Values for Business Decisions

The ability to quickly and accurately calculate the maximum value of a dataset using the MAX function in Power BI is a foundational skill in business intelligence and data visualization. Identifying these peak values provides critical insight into performance ceilings, operational limitations, or exceptional results that warrant further investigation. Whether you are benchmarking sales figures, optimizing supply chain logistics, or analyzing application usage, knowing the highest recorded metric is vital for setting realistic future goals.

By mastering the simple yet powerful syntax of the MAX function within the DAX framework, users can transform raw tabular data into concise, dynamic, and actionable reports. The provided example demonstrated the clear process--from defining the measure to visualizing the final result using the Card visual--highlighting how effortlessly Power BI handles core aggregation tasks.

Remember, the power of calculated measures ensures that your reports are not static. As new data flows into your model, the **Max Points** measure will automatically update, guaranteeing that your reports consistently reflect the true maximum performance recorded. For further learning on related functions and their usage, consult the official documentation for the **MAX** function in DAX.

The following tutorials explain how to perform other common tasks in Power BI: