

# Learning to Calculate Maximum Values with DAX in Power BI

Authored by  
**Mohammed looti**

November 12, 2025

## RECOMMENDED CITATION

Mohammed looti (2025). *Learning to Calculate Maximum Values with DAX in Power BI*. PSYCHOLOGICAL STATISTICS. Retrieved from <https://statistics.arabpsychology.com/?p=17552>

## Introduction to Calculating Maximums in Power BI

Identifying extremes within a dataset is a fundamental requirement in business intelligence, and calculating the maximum value is a core function used across all analytical platforms. Whether you are tracking the highest monthly revenue, the peak operational efficiency, or, as demonstrated here, the maximum score achieved by an individual, defining these maximums provides critical insight into performance ceilings. Within the Microsoft [Power BI](#) environment, these essential calculations are executed through the powerful formula language known as Data Analysis Expressions, or [DAX](#). Mastering the implementation of the maximum calculation is crucial for building accurate, responsive, and insightful data models.

The determination of a maximum value is typically achieved by creating a [Measure](#). Measures are distinct from calculated columns in that they perform aggregation dynamically, based on the specific context (filters, slicers, or visual axes) applied in the report view at the time of execution. This dynamic capability makes measures the industry standard for numerical aggregations, including sums, averages, and maximums. By defining the maximum score as a measure, the resulting value is guaranteed to adjust automatically if the user interacts with the report, such as applying a filter to see the maximum points scored only within a specific geographical region or time period.

To calculate the maximum value present within a single column using [DAX](#), we utilize the specific **MAX** function. This function is designed to efficiently scan all non-blank values within the designated column and subsequently return the largest numerical, textual, or date value it encounters. Its syntax is highly streamlined, requiring only the fully qualified name of the column you wish to evaluate. This simple structure provides the cleanest and most performant method for extracting peak performance indicators directly from your source data.

The correct syntax in [DAX](#) for calculating the max value in a column is as follows:

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

In this specific instance, the expression creates a new measure named **Max Points**. This measure calculates the absolute maximum value found among all entries in the **Points** column within the table designated as **my\_data**. We will now proceed with a detailed, practical demonstration showing exactly how to implement this formula within the [Power BI](#) Desktop application, transforming raw data into a usable metric.

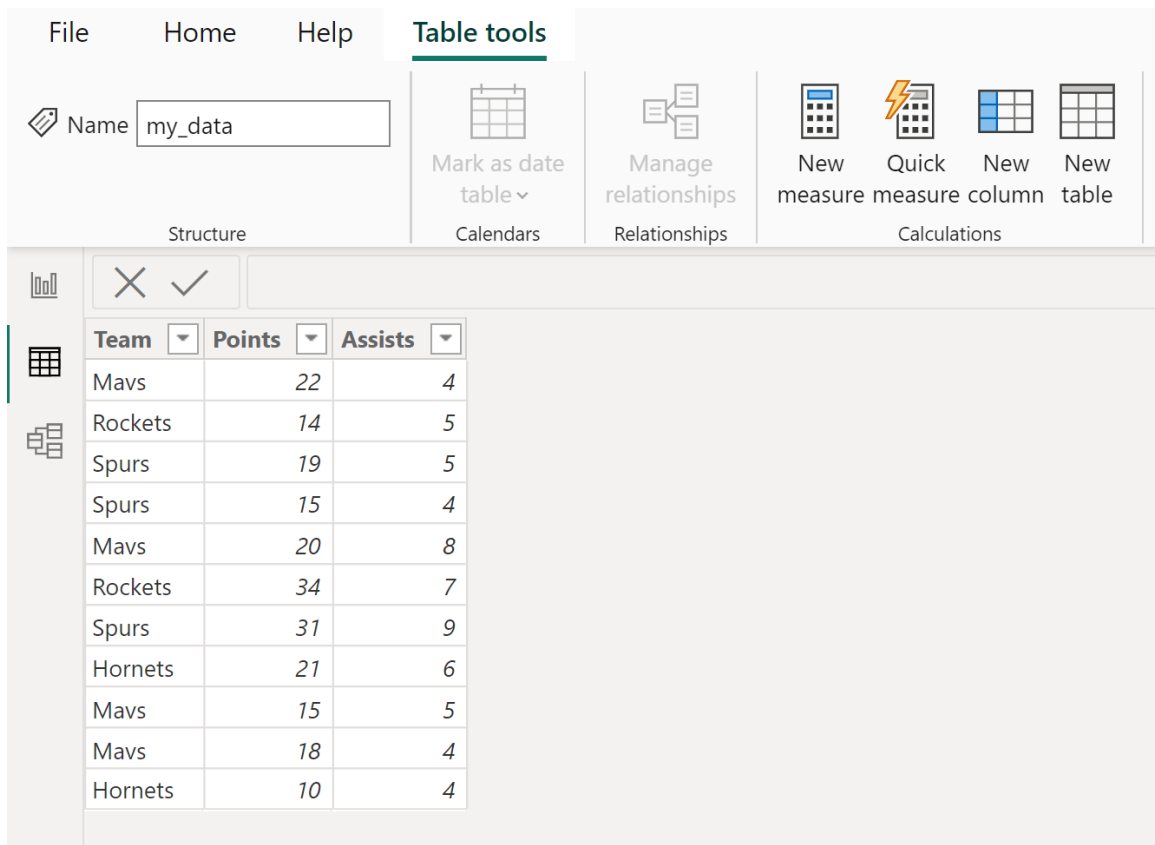
### Step-by-Step Example: Defining the Data Context

A successful data analysis project in Power BI is predicated on a well-defined and structured data

model. Before attempting to implement any DAX function, we must first confirm the context and structure of our data. For the purposes of this tutorial, we will work with an established table loaded into the Power BI environment, which we have named **my\_data**. This table contains vital information related to athletic performance, specifically tracking the **Team** affiliation of various basketball players and the total **Points** they have recorded.

It is crucial to verify the data type of the column intended for maximum calculation. The **Points** column must contain numerical values, as the standard [MAX function](#) operates exclusively on scalar values. Our immediate objective is to locate the single highest score recorded across all players listed in this comprehensive dataset, irrespective of their specific team. This resulting maximum value will serve as a foundational benchmark for subsequent comparative analyses.

The following visualization provides a clear overview of the source data table, **my\_data**, which includes details about points scored by basketball players across various teams:



The screenshot shows the Power BI interface with the 'Table tools' ribbon active. The 'Name' field is set to 'my\_data'. The ribbon includes options for 'Mark as date table', 'Manage relationships', and 'Calculations' (New measure, Quick measure, New column, New table). Below the ribbon, a table is displayed with the following data:

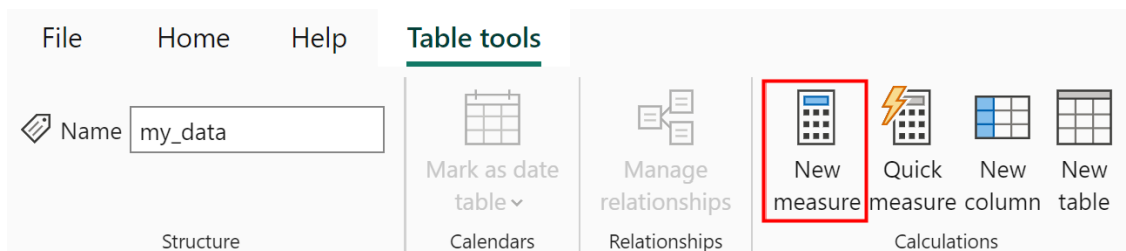
Team	Points	Assists
Mavs	22	4
Rockets	14	5
Spurs	19	5
Spurs	15	4
Mavs	20	8
Rockets	34	7
Spurs	31	9
Hornets	21	6
Mavs	15	5
Mavs	18	4
Hornets	10	4

With the data context established, our next immediate step is to define the calculation that extracts the maximum value found in the **Points** column. This calculation must be embedded into the data model as a reusable measure that can be deployed across multiple visualizations and report pages without requiring recalculation.

## Implementing the MAX Measure via Power BI Desktop

The procedure for integrating a new measure into the data model begins within the [Power BI Desktop](#) interface, utilizing the specialized modeling tools provided in the ribbon. To correctly define our maximum calculation, you must navigate to the area dedicated to data model manipulation, ensuring the calculation is categorized as a measure rather than a static column value.

To initiate the measure creation process, locate and click the **Table tools** tab positioned along the top ribbon interface. The "Table tools" contextual menu provides direct access to commands that allow structural enhancement of the data model. Once you are in the appropriate tab, click the **New measure** icon. This action immediately triggers the appearance of the formula bar, which is the designated location for inputting the [DAX](#) expression defining the measure's logical operation.

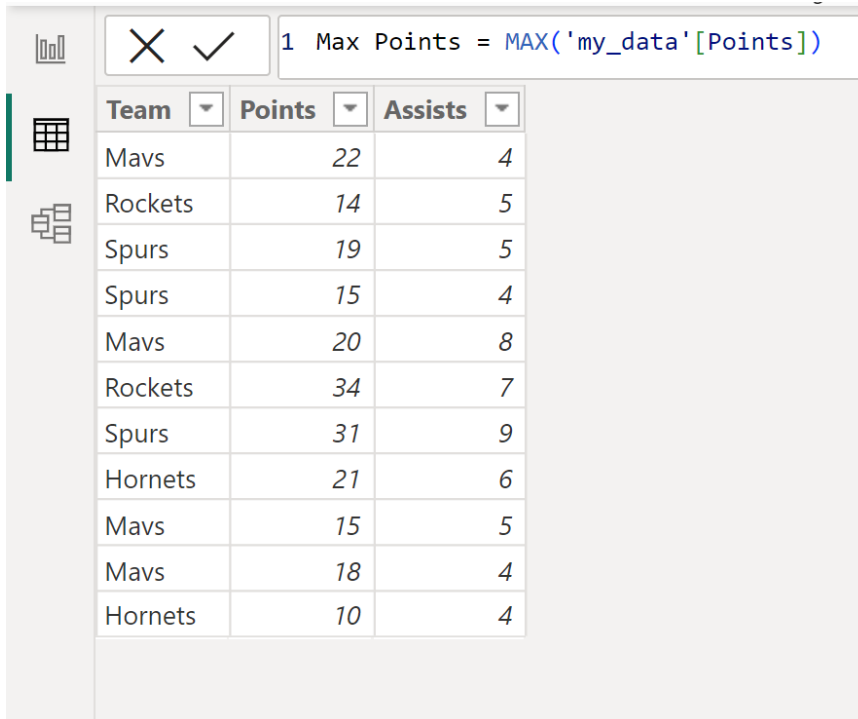


When inputting the formula, it is essential to first assign a clear, descriptive name to the measure, followed by the equals sign, and then the complete [MAX function](#) syntax. The DAX standard requires that you precisely specify the table name and the column name, separated by brackets, to avoid ambiguity. Using single quotes around the table name ('my\_data') and square brackets around the column name ( ) is crucial for ensuring the formula correctly references the data elements, regardless of naming conventions that might include spaces.

You should accurately input the following formula into the formula bar:

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

Committing this formula by pressing Enter creates the new measure, **Max Points**, which is now securely integrated into your data model. This measure holds the single maximum value extracted from the **Points** column of the **my\_data** table. The measure is instantly available for deployment within any visualization or further calculation across all pages of your Power BI report.



The screenshot shows the Power BI interface. At the top, a formula bar contains the DAX measure: `1 Max Points = MAX('my_data'[Points])`. Below the formula bar is a table with three columns: Team, Points, and Assists. The table contains 12 rows of data.

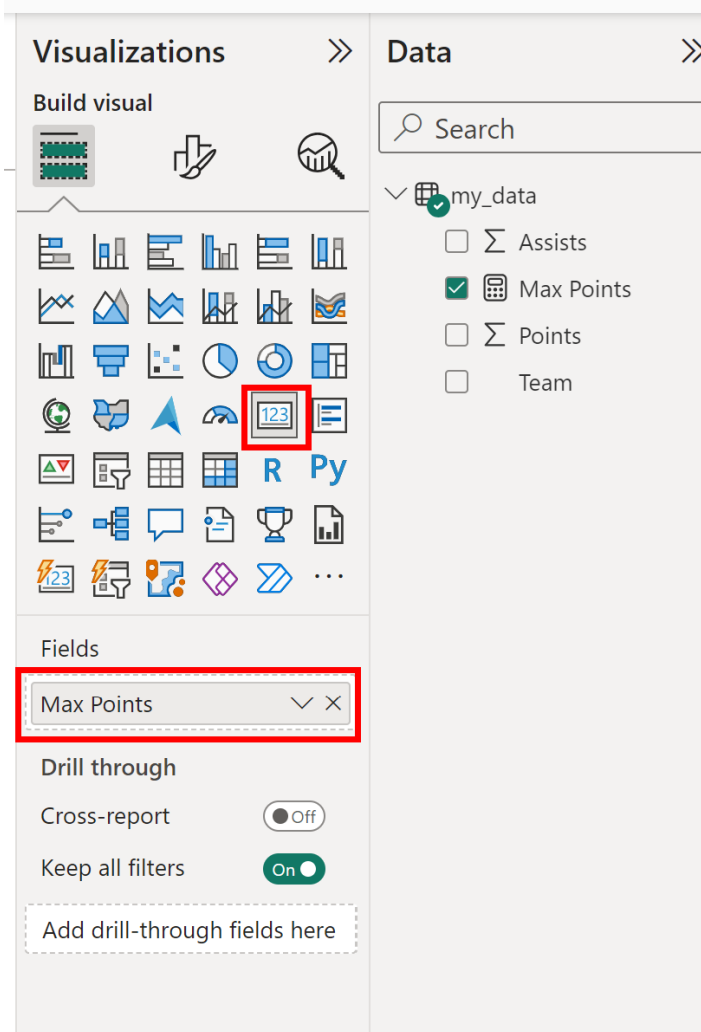
Team	Points	Assists
Mavs	22	4
Rockets	14	5
Spurs	19	5
Spurs	15	4
Mavs	20	8
Rockets	34	7
Spurs	31	9
Hornets	21	6
Mavs	15	5
Mavs	18	4
Hornets	10	4

## Visualizing the Calculated Maximum Value

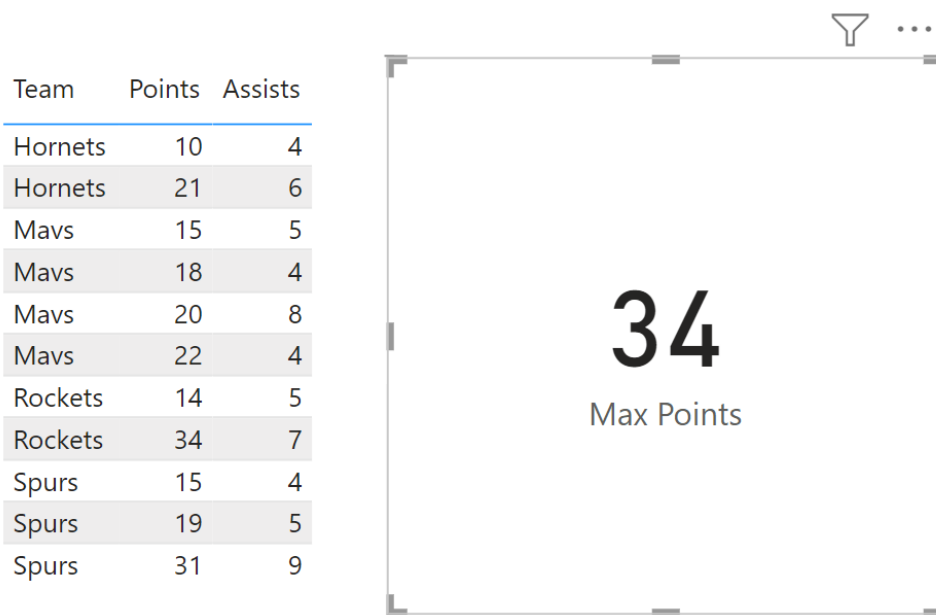
The final phase of any analytical process involves clearly communicating the results to stakeholders. After successfully calculating and storing the maximum value as a measure, the next logical step is to display this result in an easily digestible format. The preferred visualization method in [Power BI](#) for showcasing a single, aggregated numerical result is the **Card** visualization. This element is specifically optimized for highlighting key performance indicators (KPIs) and summary metrics, drawing immediate attention to critical figures.

To display the measure, you must first navigate to the [Report View](#) in Power BI Desktop, which serves as the interactive canvas for report design. Within the Report View, direct your attention to the **Visualizations** panel located on the right-hand side of the screen. Locate the icon representing the **Card** visualization and click it to introduce a new, blank card element onto your report canvas.

With the Card visualization successfully placed and selected, the final action involves linking the calculated result to the visual element. Open the **Fields** pane, which catalogs all measures, tables, and columns available in your model, and locate the newly created **Max Points** measure. Simply drag this measure from the Fields pane directly into the **Fields** well (or area) of the selected Card visualization. Power BI will instantaneously process the measure and render the result, displaying the single highest point total identified by the DAX calculation prominently on the screen.



Upon completion of these steps, the report will feature the following card visualization, which provides an immediate and unambiguous display of the maximum value found in the **Points** column of the underlying data table:



A quick glance at the resulting card confirms that the maximum value recorded in the **Points** column is precisely **34**. This visualization method provides the most effective way to communicate peak performance data efficiently and reliably to any audience interacting with the report.

## Advanced Considerations: MAX vs. MAXX in DAX

While the basic [MAX function](#) is ideally suited for determining the maximum value within a single, existing column, complex modeling scenarios in [DAX](#) often necessitate the use of its iterative counterpart: the **MAXX** function. It is imperative that advanced analysts understand the critical functional differences between these two functions, as misapplication can lead to erroneous aggregation results, especially when calculations require evaluation on a row-by-row basis before aggregation.

The standard **MAX** function operates as a simple aggregate. It takes a column reference, examines all stored values within that column (respecting the current filter context), and returns the single largest value. It is strictly column-based and does not involve iterating over table rows. Conversely, the **MAXX** function is an iterative function, often categorized as an X-function. It requires two main arguments: the table over which to iterate, and an expression to evaluate. **MAXX** executes the provided expression for every single row of the specified table and then returns the maximum result derived from that set of row-level evaluations. This iterative capability is vital when calculating the maximum of a derived metric, such as the maximum discount percentage applied per transaction, which must be calculated for each row individually before the maximum is selected.

For straightforward aggregation, such as finding the maximum value in our already calculated **Points** column, using **MAX** is both the most direct and most resource-efficient method. However, if the requirement were to find the maximum value of a calculated measure--for example, the maximum adjusted score (where the adjusted score is Points minus Fouls)--the **MAXX** function would become necessary. It ensures that the subtraction (Points - Fouls) is performed for every row before the maximum result is subsequently selected. Understanding and correctly applying both **MAX** and **MAXX** is a cornerstone of building flexible and highly accurate DAX data models.

## Summary and Additional Resources

Successfully calculating the maximum value in [Power BI](#) through the use of the [DAX MAX function](#) is a foundational skill for all data professionals utilizing the platform. By adhering to the standardized process of creating a new measure and precisely applying the DAX syntax, analysts can rapidly generate reliable key performance indicators that effectively highlight peak values within their datasets. It is important to always remember that DAX measures provide dynamic results, adjusting instantly to report filters and slicers, which ensures superior accuracy compared to static calculations performed outside of the data model context.

**Note:** For advanced scenarios, including the behavior of the **MAX** function when dealing with non-numeric data types (such as text or dates) and its complex interactions within filtered contexts, it is highly recommended to consult the official documentation for [DAX](#). This ensures that you are able to leverage the function's capabilities to their fullest extent, regardless of the complexity of your data model.

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