

# Learning Power BI: How to Use Conditional Formatting in Bar Charts

Authored by  
**Mohammed loot**

November 12, 2025

## RECOMMENDED CITATION

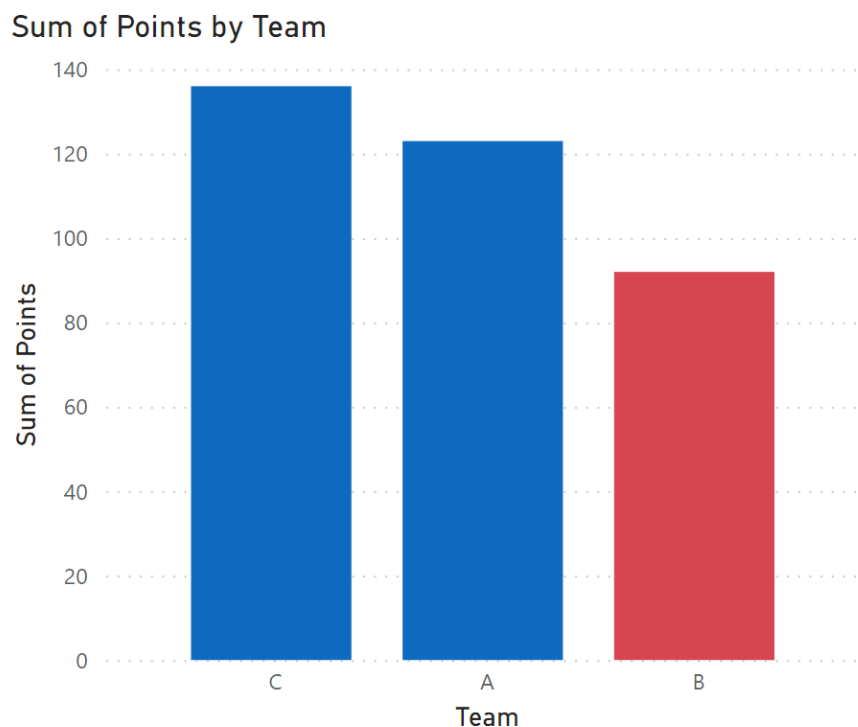
Mohammed loot (2025). *Learning Power BI: How to Use Conditional Formatting in Bar Charts*. PSYCHOLOGICAL STATISTICS. Retrieved from <https://statistics.arabpsychology.com/?p=17658>

## The Importance of Conditional Formatting in Data Visualization

In modern [data visualization](#), merely displaying data is often insufficient for driving business decisions. Effective reporting requires techniques that highlight critical insights immediately, guiding the user's attention to areas that require action or closer scrutiny. This is where [conditional formatting](#) becomes an indispensable tool within reporting platforms like [Power BI](#). Conditional formatting allows analysts to dynamically apply visual styles--such as color changes, font adjustments, or icon overlays--based on the underlying data values.

A common requirement involves applying distinct coloring to bar charts. For instance, you might need bars representing performance metrics above a certain target threshold to be colored green, signifying success, while those falling below that threshold are colored red, indicating risk or underperformance. This immediate visual feedback transforms a standard static chart into a highly dynamic and informative dashboard element. The following demonstration walks through the precise steps necessary to implement this powerful feature, ensuring that bars exceeding a specified numerical value adopt one color, while bars below that value adopt another, creating clear boundaries in your analysis.

Consider the following common scenario where different colors are used to differentiate successful results from suboptimal ones. This specific technique leverages custom rules rather than standard color gradients to establish hard cut-offs based on a defined numerical benchmark.

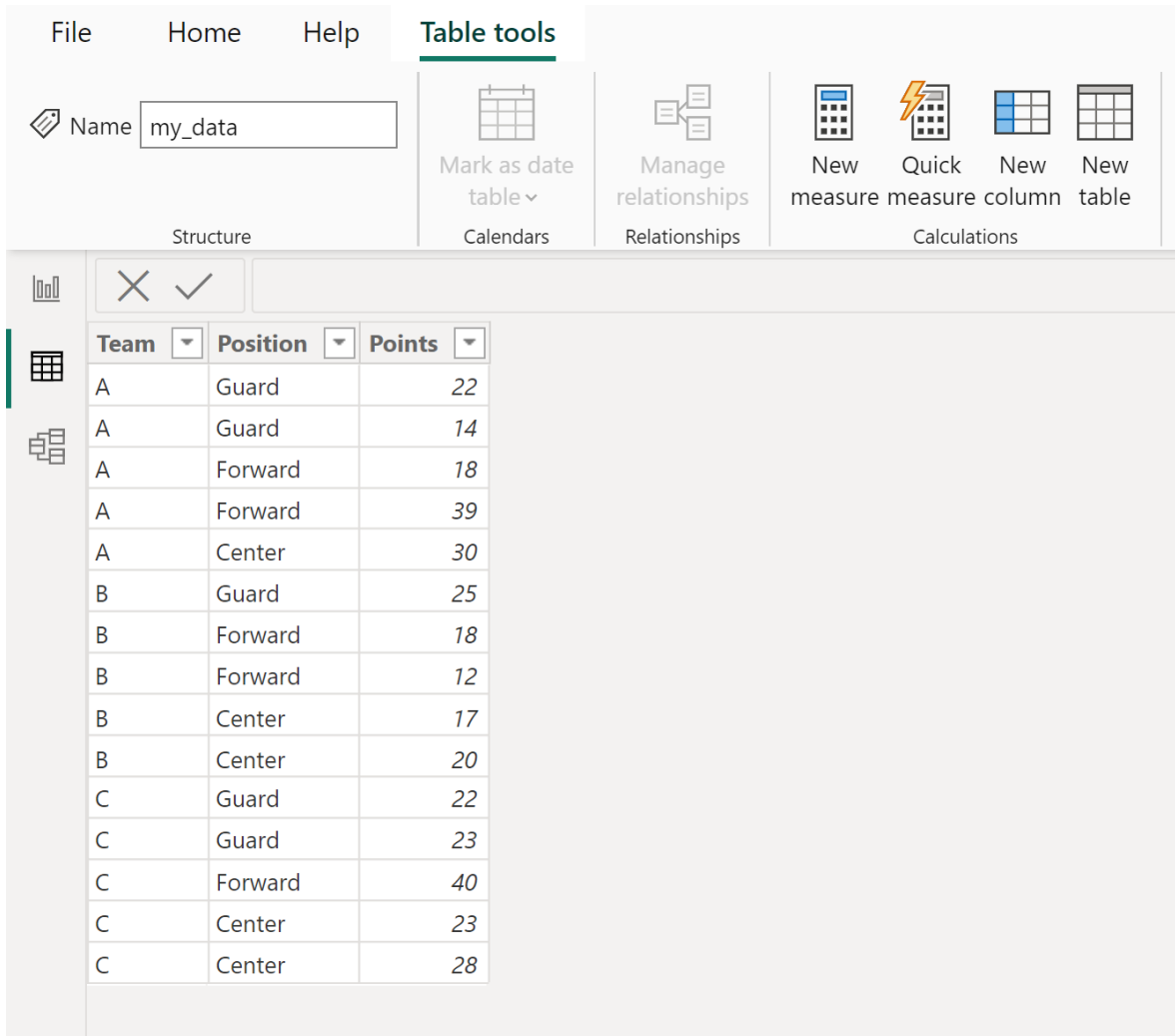


Fortunately, the process for setting up these explicit visual rules in [Power BI](#) is intuitive and highly effective. We will now proceed with a detailed, step-by-step example using a sample dataset to illustrate the implementation process from initial setup to final visualization enhancement, focusing specifically on achieving a clear binary color distinction.

## **Preparing Your Data Environment in Power BI**

Before applying any visualization techniques, we must ensure our data is properly loaded and modeled within the [Power BI](#) Desktop environment. For this example, we will utilize a dataset containing performance statistics for various basketball players. This dataset includes key fields such as the player's team affiliation and the total points they have scored. Understanding the structure of your source data is the foundational step for any successful report creation, as it dictates how fields should be utilized in the visual axes.

Suppose we are working with the following sample dataset already loaded into our Power BI model. This table provides a clear view of the teams involved and the raw point totals associated with them, serving as the basis for our forthcoming bar chart visualization. This data structure facilitates the aggregation necessary to compare team performance.

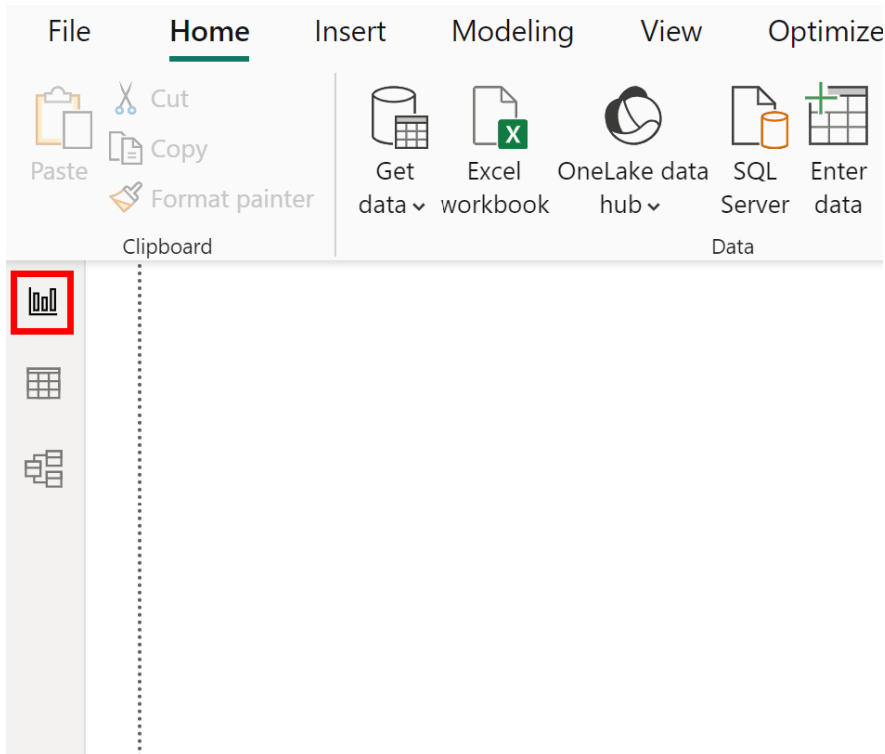


The screenshot shows the 'Table tools' ribbon in Power BI Desktop. The ribbon is divided into three main sections: 'Structure', 'Relationships', and 'Calculations'. The 'Structure' section includes a 'Name' field with the value 'my\_data'. The 'Relationships' section includes 'Mark as date table', 'Manage relationships', and 'Relationships'. The 'Calculations' section includes 'New measure', 'Quick measure', 'New column', and 'New table'. Below the ribbon, a table is displayed with the following data:

Team	Position	Points
A	Guard	22
A	Guard	14
A	Forward	18
A	Forward	39
A	Center	30
B	Guard	25
B	Forward	18
B	Forward	12
B	Center	17
B	Center	20
C	Guard	22
C	Guard	23
C	Forward	40
C	Center	23
C	Center	28

Our primary objective is to aggregate this data to visualize the cumulative sum of points scored by each team. This aggregation step is crucial because bar charts typically display summarized metrics (like sums or averages) rather than individual raw rows. We aim to compare the overall performance of different teams against each other using a clear, vertical comparison structure provided by the column chart.

To begin the visualization process, navigate to the primary reporting interface. Click the **Report View** icon, which is prominently located on the left-hand navigation pane of the Power BI Desktop application. This action shifts the interface from the Data or Model view back to the canvas where visual elements are constructed and arranged for report consumption.



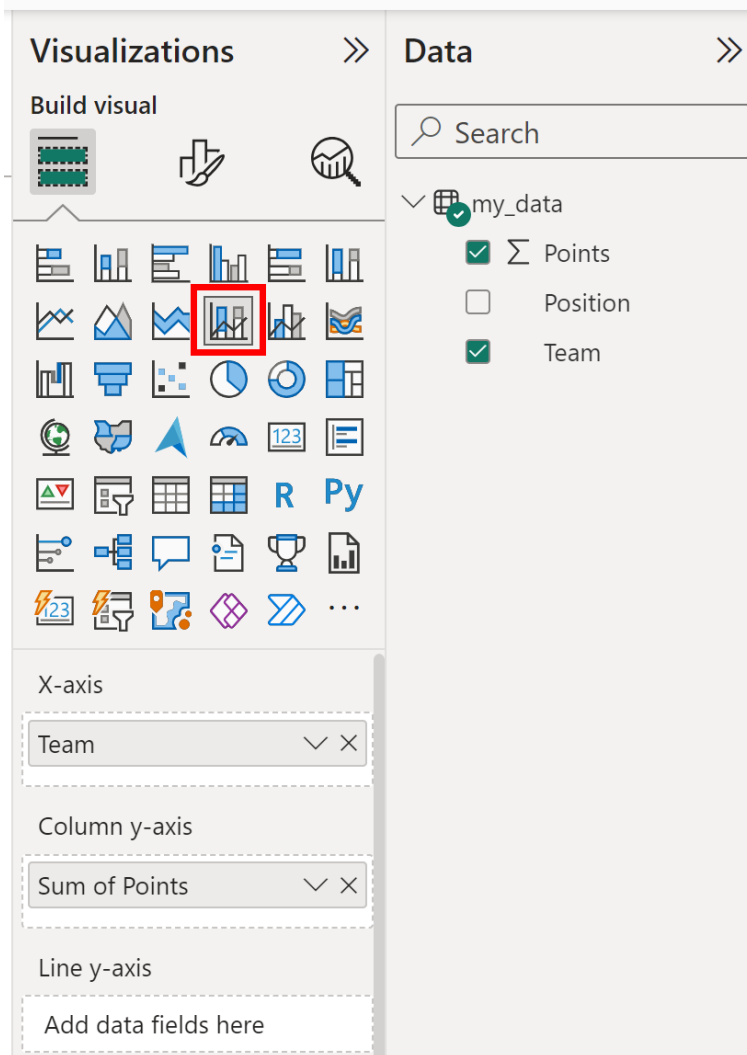
## Constructing the Initial Bar Chart Visual

Once in the [Report View](#), the next step involves selecting the appropriate visualization type from the available options. For this specific demonstration, we will select the **Line and stacked column chart** visual found within the [Visualizations Pane](#) on the right side of the screen. Clicking this icon places an empty visual container onto the report canvas, ready for data field assignment.

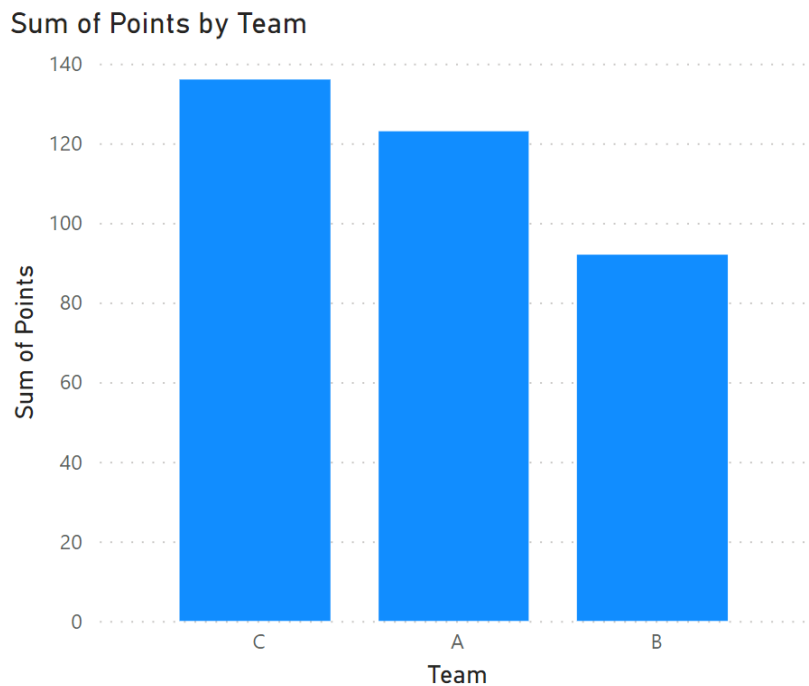
After the visual is placed, we must assign the relevant data fields to the appropriate axes. Proper field assignment dictates how the data is displayed and aggregated. For our objective--visualizing total points per team--we will make the following configuration choices within the Visualizations pane, ensuring the correct data roles are fulfilled:

Assign the **Team** field to the **X-axis** slot. This category field will determine the distinct bars displayed horizontally along the category axis.

Assign the **Points** field to the **Column y-axis** slot. Power BI automatically defaults to summarizing numerical fields (specifically using the SUM function), so this placement calculates the total points for each team category defined on the X-axis.



Upon assigning these fields, Power BI renders the initial visualization. This resulting chart visually represents the total points aggregated across all players belonging to each respective team. At this stage, the chart uses the default color scheme defined by the report theme, providing a baseline comparison of team performance without any specific emphasis on targets or thresholds.

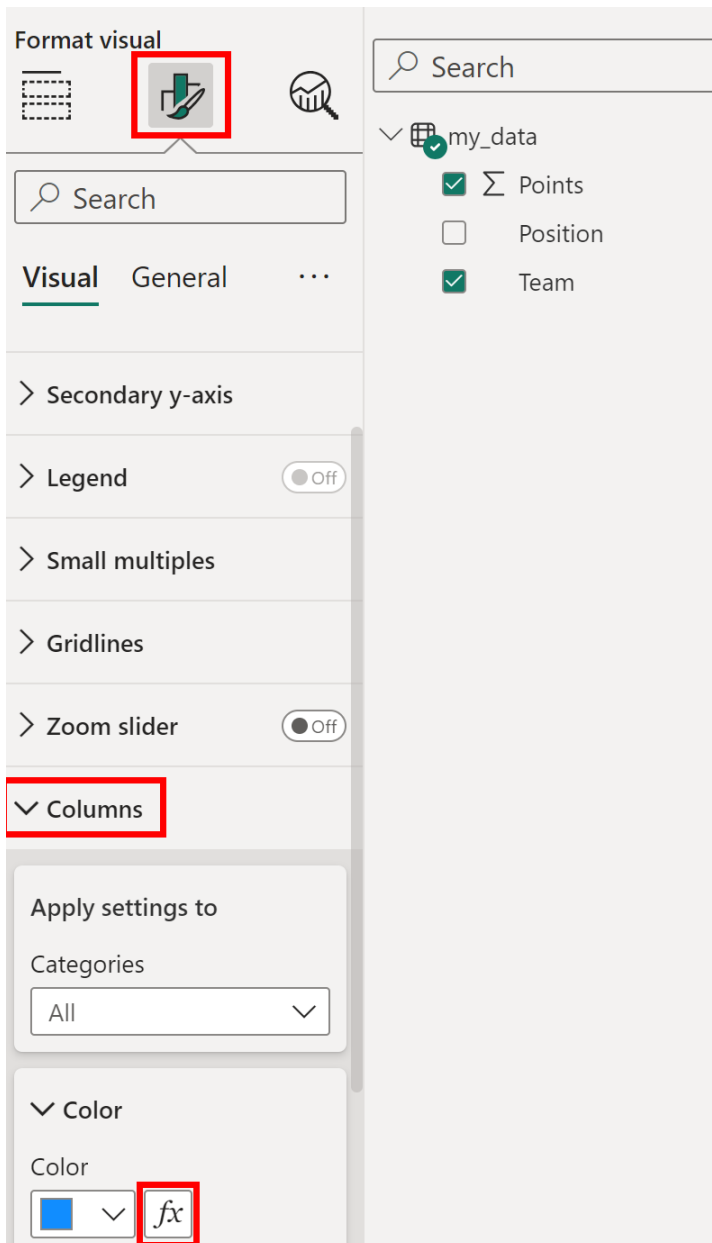


The visual is now complete in terms of data presentation. However, to enhance its interpretability, we must introduce the [conditional formatting](#) rules that will make the high-performing teams stand out instantly from the lower-performing ones based on our defined metric.

## Implementing Advanced Color Rules via Conditional Formatting

The core purpose of this exercise is to apply binary conditional formatting: bars with a cumulative value of 100 or greater must be colored **blue**, while bars with a value less than 100 must be colored **red**. This threshold of 100 serves as a critical performance benchmark, dividing our teams into two distinct performance groups. To achieve this separation, we need to access the advanced formatting controls within the visual settings.

With the chart selected on the canvas, navigate to the **Format your visual** tab (represented by the paintbrush icon) in the [Visualizations Pane](#). We need to locate the specific color controls for the columns. Expand the dropdown arrow associated with the **Columns** section. Within this section, instead of choosing a static color, locate and click the **Conditional Formatting** (labeled as *fx*) icon. This icon opens a powerful dialog box allowing rule-based styling to be applied to the column colors.



The Conditional Formatting dialog box requires careful configuration to establish the desired hard cut-off between the red and blue zones. While the default setting might suggest a smooth gradient, we can manipulate the minimum and maximum values to force an abrupt color change exactly at our threshold (100 points). The key to this technique is setting the minimum color to end immediately *before* the target value and the maximum color to begin immediately *at* the target value, thereby creating a sharp boundary.

We will configure the rules using the following settings within the Conditional Formatting dialog, ensuring the logic adheres strictly to our goal of differentiating performance at the 100-point mark:

**Format style:** We select **Gradient**. Even though we aim for a binary color scheme, using the

gradient option allows us to define two distinct colors based on custom minimum and maximum values that butt up against the threshold.

**What field should we base this on:** This must be set to the measure being visualized, which is the **Sum of Points** (or simply **Points**). This ensures the color logic evaluates the height of the bar itself.

**Minimum:** This needs to cover all values below 100. We set the type to **Custom** value and input 99, selecting the color **Red**. Any value up to and including 99 will be colored red, signaling below-target performance.

**Maximum:** This defines the starting point for the successful color. We set the type to **Custom** value and input 100, selecting the color **Blue**. Any value of 100 or greater will transition instantly to blue, signifying target achievement.

This precise configuration ensures that the color transition happens exactly between 99 and 100, effectively creating the binary threshold we specified without any intermediate gradient colors.


### Color - Categories

Format style  
Gradient

What field should we base this on? Summarization How should we format empty values?  
Sum of Points Sum As zero

Minimum Maximum  
Custom 99 Custom 100

Add a middle color

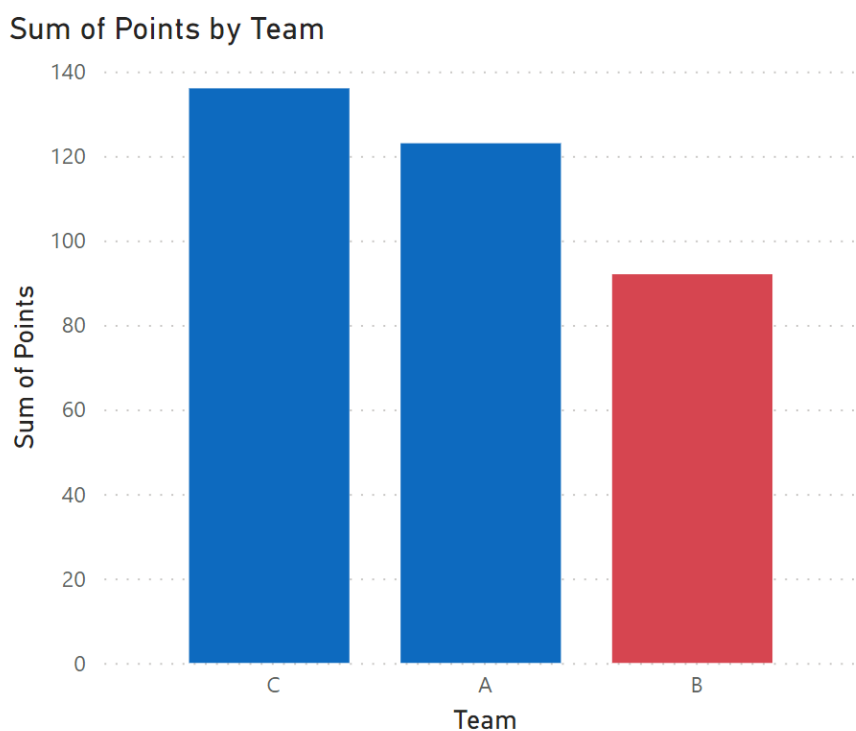


[Learn more about conditional formatting](#) OK Cancel

## Reviewing the Enhanced Visualization and Next Steps

Once the settings are confirmed by clicking **OK** in the Conditional Formatting dialog, the changes are immediately applied to the visual on the report canvas. The bar chart transforms instantly: teams whose aggregated points are 100 or higher are now colored blue, clearly indicating they have met or exceeded the performance benchmark. Conversely, teams that scored less than 100 points remain prominently marked in red, instantly drawing the user's eye to areas requiring further investigation.

The resulting visualization is far more impactful than the original default chart. This use of [conditional formatting](#) dramatically improves the speed at which users can glean insights from the data, providing an immediate visual comparison against the defined target.



We have successfully applied sophisticated conditional formatting to our bar chart, basing the color scheme purely on the numerical values of the aggregated bars. This technique is highly adaptable and can be applied to various numerical metrics across almost all visual types in [Power BI](#), including tables, matrices, and other chart types. Mastering this skill is fundamental to building professional and action-oriented dashboards that effectively communicate key performance indicators.

For analysts looking to further expand their Power BI skillset, understanding how to apply conditional formatting based on different data types, such as text fields or date ranges, is the

logical next step. These advanced capabilities unlock full control over the visual storytelling aspect of data reporting, enabling the creation of truly dynamic and engaging reports.

## **Additional Resources for Power BI Mastery**

To deepen your expertise in advanced report design and formatting within Power BI, explore tutorials on related common tasks. These resources cover essential skills necessary for creating comprehensive and visually appealing reports that move beyond standard chart presentations:

[Power BI: How to Apply Conditional Formatting to Text Column](#)