How to Easily Remove Blank Rows in Power BI

Authored by

mohammed looti

January 12, 2026

The Critical Role of Data Cleansing in Power BI

Data quality is paramount when performing business intelligence tasks. Successfully analyzing and visualizing data within Power BI hinges on the cleanliness and accuracy of the source information. One of the most common data integrity issues encountered by analysts is the presence of blank or null rows. These rows, which often result from flawed data extraction processes, incomplete user input, or errors during data merging, can severely skew calculations, introduce inaccuracies into measures, and complicate the overall data modeling process.

Fortunately, Power BI provides robust, intuitive tools for data cleansing. Removing blank rows is a core requirement for ensuring the reliability of any data set. While it is technically possible to remove blank rows using simple column filtering within the visual canvas, the most efficient and recommended approach is to utilize the comprehensive capabilities offered by the **Power Query Editor**. This tool allows for permanent, repeatable transformations that are applied every time the data is refreshed.

This detailed guide will walk you through the precise steps required to identify and eliminate entirely blank rows from your tables, transforming a messy data set into a structured source ready for high-fidelity reporting and visualization. Understanding this foundational data cleansing technique is essential for any professional working extensively with Power BI Desktop or Service.

The Power Query Advantage: Automated Row Removal

As established, the most straightforward and reliable method for achieving thorough data cleanliness is by leveraging the **Remove Blank Rows** function directly within the Power Query Editor. Power Query (also known as Get & Transform Data) is the specialized ETL (Extract, Transform, Load) tool integrated into Power BI. It allows users to manipulate data structure, types, and content before it is loaded into the Data Model. This approach is superior to simple front-end filtering because it permanently modifies the data source structure applied to the semantic model.

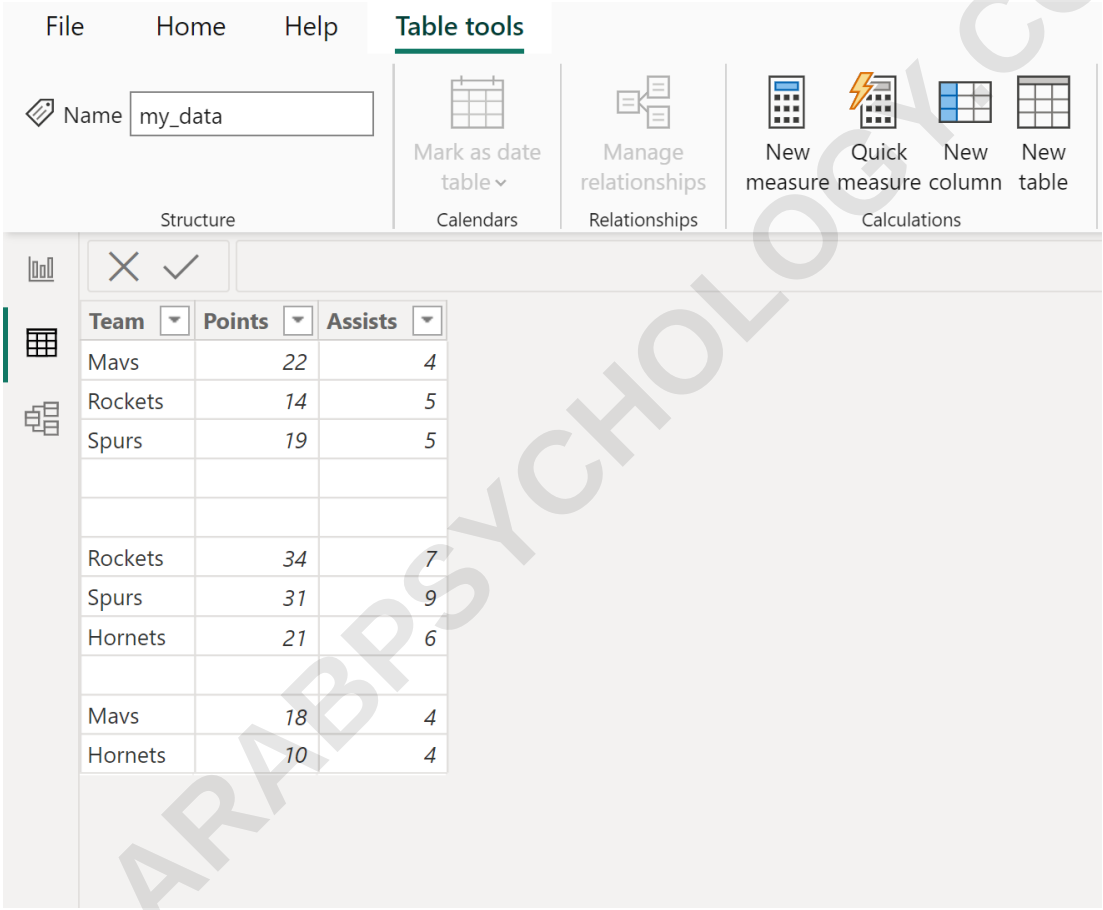
When dealing with large or complex imports, manually filtering individual columns can be time-consuming and error-prone. The **Remove Blank Rows** feature automates the detection of rows where all fields contain null values or are empty strings, efficiently purging them from the table with a single click. This guarantees consistency and saves significant time during the data preparation phase.

The following practical example illustrates how to utilize this essential feature, using a hypothetical data table containing player statistics. We will demonstrate how blank rows, which compromise the integrity of the statistics, are swiftly eliminated using the built-in Power Query functionalities.

Illustrative Example: Data Set Preparation

Imagine a scenario where we have imported a data set intended to track various basketball players and their corresponding performance metrics. However, due to data entry mistakes or systemic integration issues, several rows are entirely blank, meaning every cell in those rows contains no information, as highlighted in the preview below.

These blank entries must be removed before any meaningful aggregation or analysis can take place. If they remain, they might cause issues with calculations, particularly counts or averages, and unnecessarily increase the size of the data model.



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

Carefully observe the table above. You can clearly identify the rows that contain blank values in every visible column. Our objective is to streamline this table, retaining only the legitimate player records. We will now proceed into the transformation environment to initiate the cleansing process.

Accessing the Power Query Editor Interface

To begin the process of data transformation, you must first transition from the main Power BI desktop to the dedicated Power Query Editor interface. This is accessed through the primary

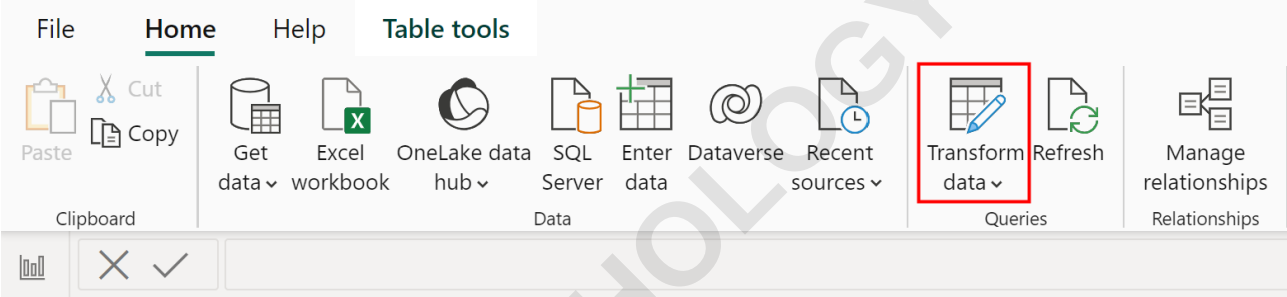
ribbon within the Power BI Desktop application. This action effectively pauses the data visualization process and shifts focus entirely to data preparation.

Follow these navigational steps precisely:

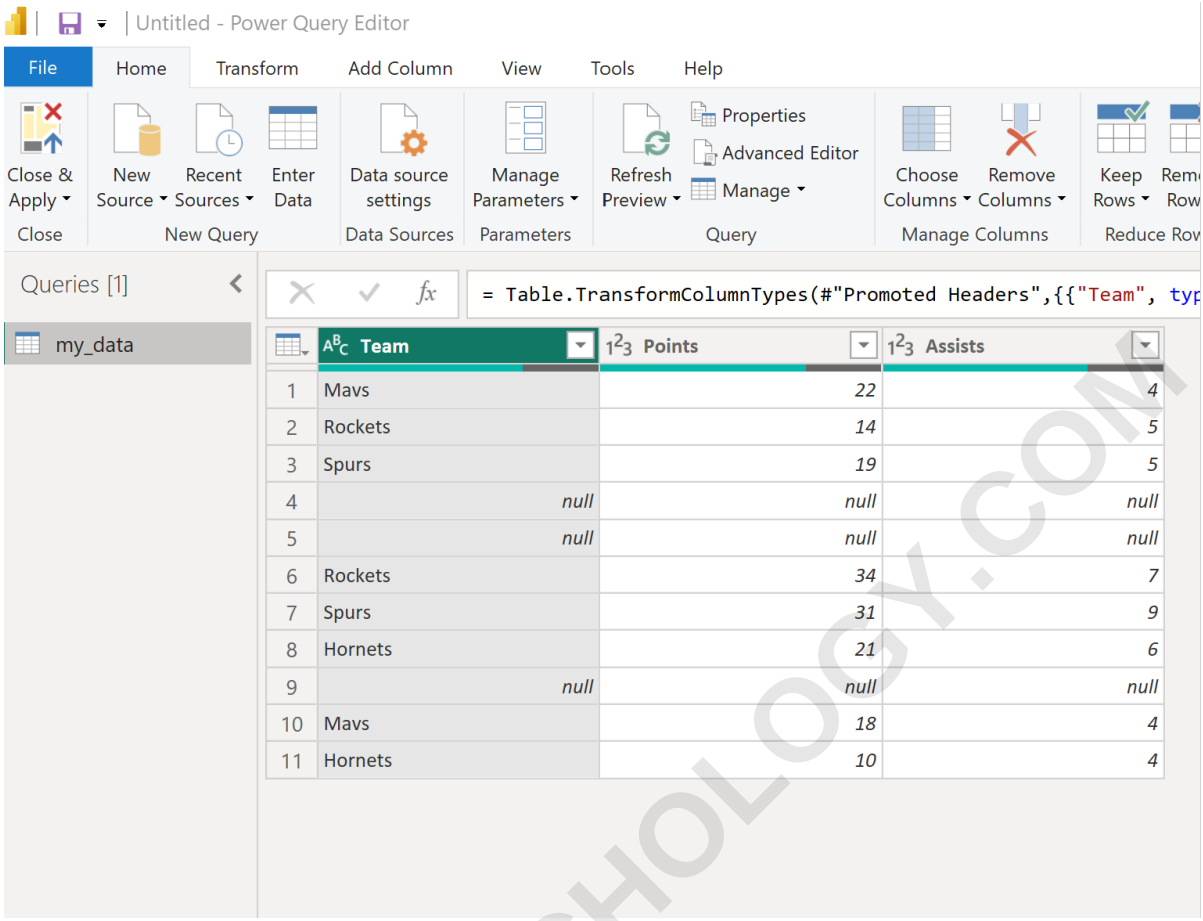
Navigate to the **Home** tab located at the top of the Power BI Desktop application ribbon.

In the external data section, locate and click the **Transform data** icon. This action typically opens a new, dedicated window containing the Power Query interface. If the data set has not been loaded, this step may initiate the connection wizard.

The image below illustrates the location of the **Transform data** button within the ribbon interface, which is the gateway to all advanced data manipulation functions.



Upon clicking, the system launches the **Power Query Editor** window. This environment is where all data preparation steps are recorded as applied steps. These steps are crucial because they ensure that the data transformation is repeatable and automatically executed upon future data refreshes, guaranteeing consistent data quality over time.



Once inside the editor, verify that the correct table (in this example, the table containing basketball player data) is selected in the Queries pane on the left-hand side. The data preview pane in the center should display the original data, including the undesirable blank rows we aim to eliminate.

Executing the Automated Removal of Blank Entries

With the relevant table selected within the Power Query Editor, the next step involves using the specific feature designed for this data cleansing task. This function is conveniently located within the Power Query ribbon, under the primary control group dedicated to row management.

To successfully remove all completely blank rows:

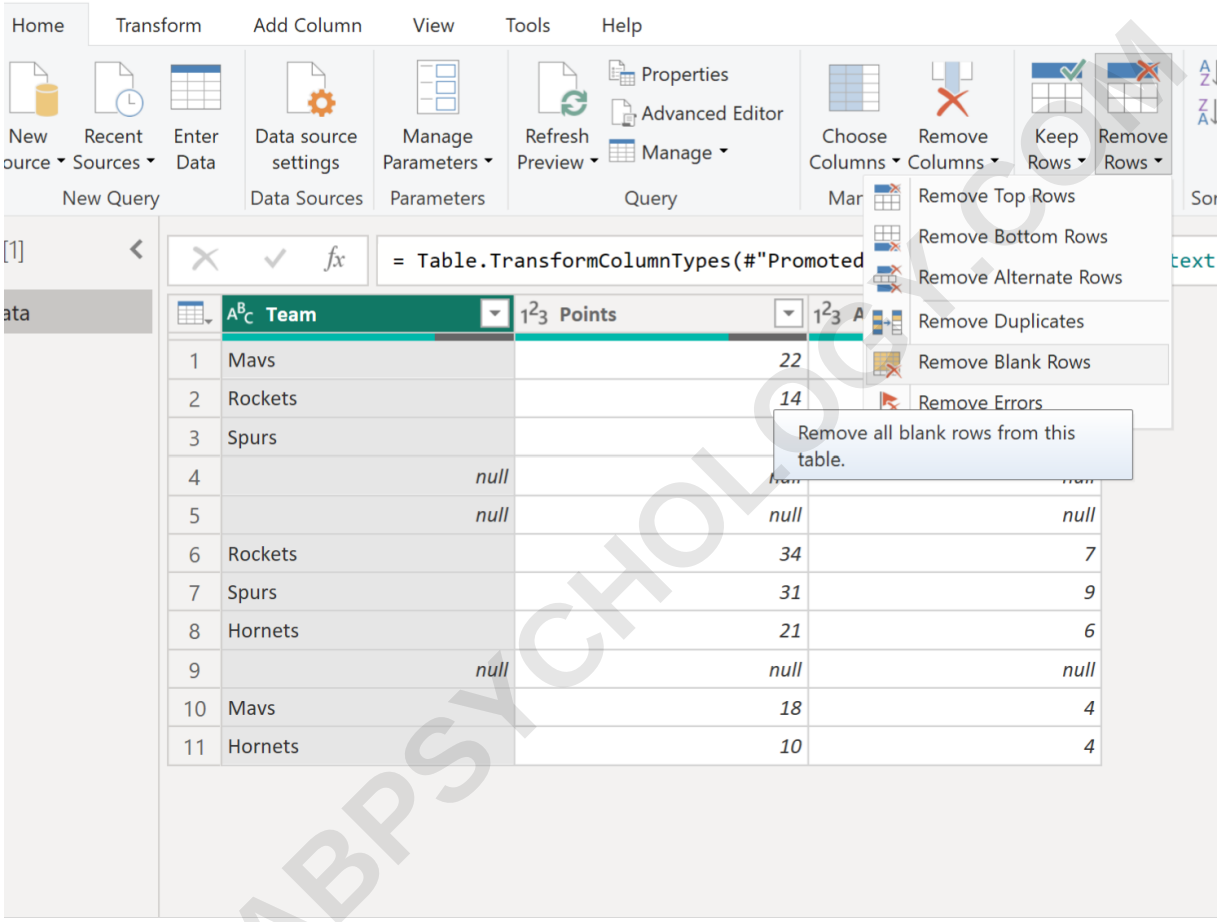
Ensure you are on the **Home** tab within the Power Query Editor ribbon.

Locate the **Reduce Rows** group, typically positioned toward the left side of the ribbon.

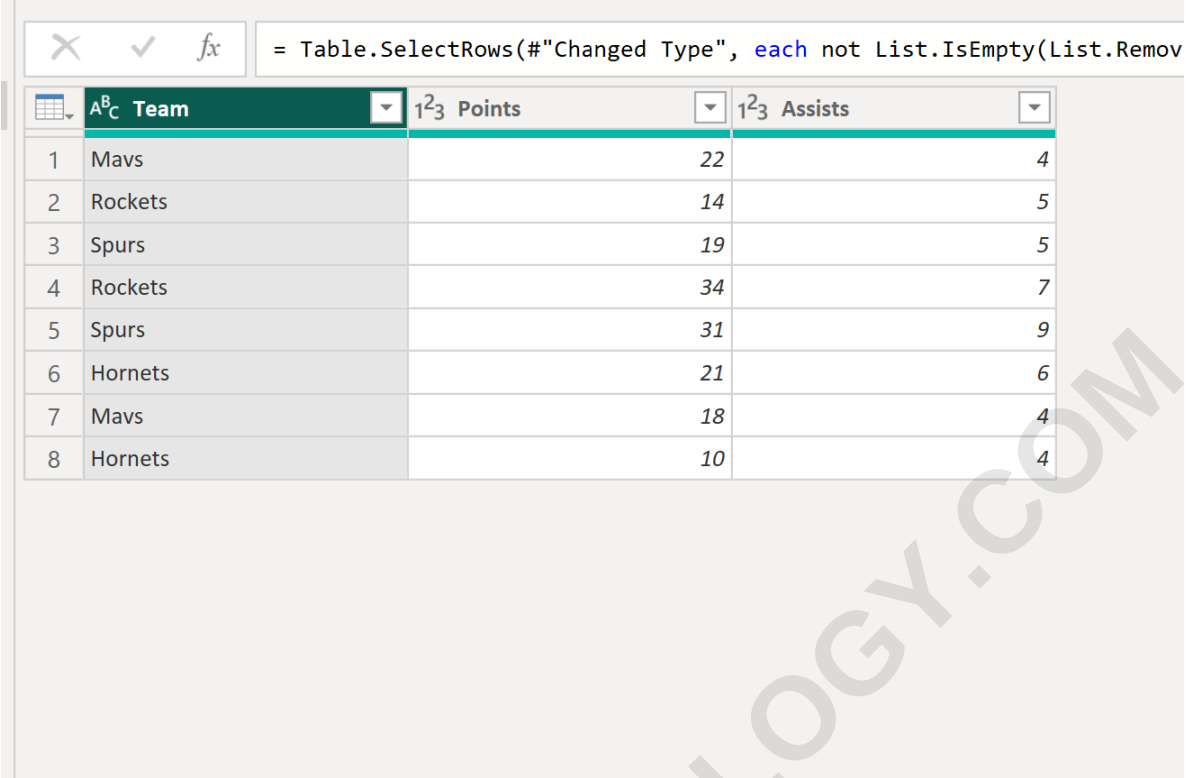
Click on the **Remove Rows** command. This action will reveal a drop-down menu containing several options for row deletion.

From the drop-down menu, select **Remove Blank Rows**.

This specific command instructs the M engine underlying Power Query to evaluate every row in the selected table. Any row where all cells, across all columns, contain a null value will be instantly purged from the data view. The image below highlights where this critical option is located within the interface hierarchy.



Observe the immediate transformation in the data preview pane. All blank rows that were previously visible have vanished, leaving only the records that contain substantive data. The result is a clean, optimized data table ready for loading into the data model.



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

It is important to note that this transformation is captured as a new step in the **Applied Steps** panel on the right side of the editor. This step, named something similar to "Removed Blank Rows," can be reviewed, edited using the M language formula bar, or deleted entirely if the change needs to be reversed. This history logging provides exceptional control and auditability over the data cleansing process.

Committing Changes and Loading Clean Data

The transformation applied within the Power Query Editor is currently only staged; it has not yet been committed back to the main Power BI Data Model. To finalize the data cleansing and ensure the subsequent reports utilize the refined data set, you must save and apply the changes.

To exit the Power Query Editor and apply the changes:

Navigate to the **Home** tab within the Power Query Editor window.

Click the **Close & Apply** button, typically found on the far left of the ribbon.

Alternatively, simply attempting to close the Power Query Editor window will prompt a dialogue box asking if you wish to apply the pending changes to the underlying data model. Clicking **Yes** ensures that the data is loaded without the blank rows. This process triggers a refresh where the data is read through all the applied steps, including the "Removed Blank Rows" step.

Once the changes are successfully applied, the table within the Power BI Desktop environment will reflect the cleaned data, guaranteeing that visualizations, measures, and calculated columns are based on accurate and complete records. This transformation is persistent, meaning the blank rows will remain excluded every time the data source is updated or refreshed.

Advanced Scenario: Removing Blanks Based on Specific Columns

While the **Remove Blank Rows** feature is perfect for dealing with rows that are entirely empty, data quality issues are often more nuanced. Sometimes, a row might contain data in several columns but still be considered "blank" because a critical identifier or measure column is missing. For example, a basketball player record might have a name but no recorded points, rendering the row useless for scoring analysis. In such cases, relying on column filtering provides a more surgical approach.

If you only want to remove rows where a specific column (or subset of columns) is blank, you should use the standard column filtering mechanism within the Power Query Editor. To do this, click the filter dropdown arrow located at the top of the relevant column header (e.g., the 'Player Name' column).

In the filter menu that appears, simply uncheck the box labeled **(null)** or **(blank)**, depending on the data type and how the null values were imported. This action creates a filter step that ensures only rows containing a valid value in that particular column are retained. This technique is highly effective when the definition of a "blank row" is conditional rather than absolute across all fields. Remember that combining multiple column filters can further refine your data set to meet specific quality standards.

Best Practices for Maintaining Data Integrity

Preventing blank rows is often easier than removing them after the fact. Maintaining high data integrity requires a holistic approach that starts at the source system and continues through the ETL pipeline. Analysts should consider the following preventative measures and best practices when dealing with data importation and transformation in Power BI.

First, implement robust data validation rules at the source. If data is being entered manually, ensure fields marked as non-nullable in the database schema are strictly enforced. Second, when using external data sources like CSV or Excel, examine the import settings carefully. Ensure that blank cells are consistently interpreted as **null** by Power Query, which standardizes the values and makes filtering easier. Inconsistent representation (e.g., some blanks as null, others as empty strings) requires more complex conditional transformations.

Furthermore, always document your transformation steps. The Power Query Editor allows you to

rename and add descriptions to each step in the Applied Steps panel. A well-documented process ensures that colleagues can understand the logic behind the data cleansing, especially the decision points regarding whether to use the global **Remove Blank Rows** feature or specific column filters. Regular review of the data refresh performance can also highlight recurring blank row issues that signal deeper problems in the source data feed, prompting necessary upstream corrections.