Data Transformation with Power Query / Query Editor

Data Transformation

Data transformation is the process of converting data or information from one format to another, usually from the format of a source system into the required format of a new destination system.

Data Transformation – Why?

When existing Business model is hard to understand we use power query to Shape or Transform the data, and build a model that will be easily understandable for a Report User.

If existing Business model contains too many tables and many relationships between tables makes a reporting query very slow and not efficient. Here we use Power Query to Shape and Transform the data to build a star or snow flake schema by creating dimension tables and fact table, which is more comfortable for report development.

Transactional databases are not best option for reporting purpose because

- ✓ The model is hard to understand for a Report User.
- ✓ Too many tables and many relationships between tables makes a reporting query (that might use 20 of these tables at once) very slow and not efficient.
- ✓ Also we don't need all the transactional data to be loaded into Reporting Tools we just load whatever data we need for reports into our reporting tools.

Shape or Transform Data using Power Query

With Power BI Desktop or Query Editor or Power Query, you can connect different types of data sources, and then shape the data to meet your reporting needs.

In Power Query or Query Editor we will transform or shape the data using built-in GUI transformations in the ribbon or using M language code.

Benefits of Data Transformation

Data transformation ensures that data that enters your enterprise is usable and manageable.

It facilitates cost-efficient storage, ease of analysis for greater business intelligence, and operational efficiency.

On the flip side, storing data that has not been transformed wastes resources and creates the possibility of compliance risk because the data cannot be managed under the organization’s data governance rules.

Overview of Power Query / Query Editor

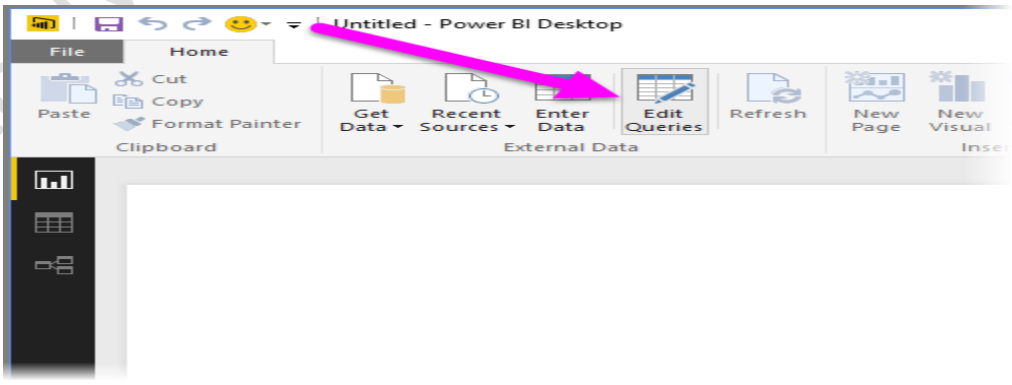
- ✓ Power Query is a Data Extraction, Transformation and Loading Engine.
- ✓ The Engine comes with a Graphical Tool and a Formula Language (M Language).
- ✓ Power Query can connect to set of data sources and read data from them for data preparation.
- ✓ Once connected to any data source, then Queries (one for each table, or entity) are listed and available for selection, viewing, and shaping.
- ✓ The Graphical Tool has list of Transformations that can be applied on a data set or Queries, and it also supports different data sources.
- ✓ Power Query graphical interface is so easy to work with that even business analyst or a power user can work with it, on the other hand Power Query M language is so powerful that can be used for complex real world challenges of data transformations.
- ✓ However, the Power Query formula language (M Language) is much more powerful than the GUI. Actually there are some features in Power Query engine that not yet has been implemented through GUI, but they are available through M Language.
- ✓ Power Query can load the result set into Power Pivot for data modeling.
- ✓ M is the formula language behind the scenes of Power Query. Everything you do in the Query Editor will be translated to an M script. M contains full list of functions that you can use. So the powerful side of Power Query is actually M. M is a functional language and it has a simple structure.
- ✓ Every data preparation steps or applied steps on Queries will be recorded and displayed in Query Editor under Applied Steps Section.

Query Editor User Interface

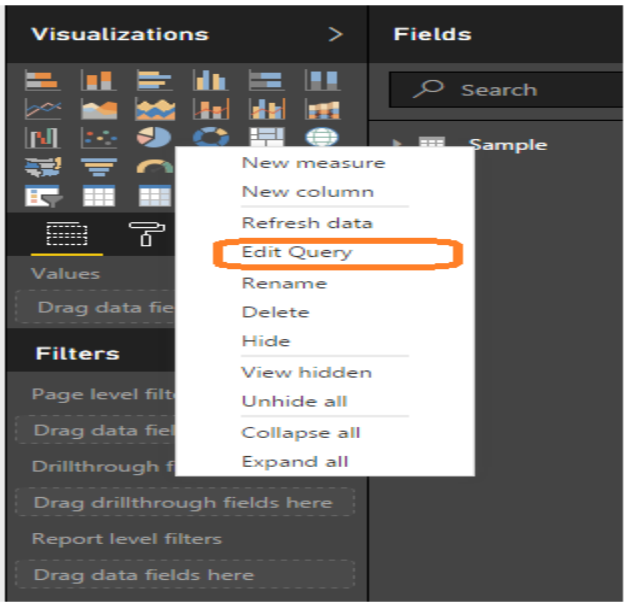
You can open Power Query Editor in three different ways

1. From Home Tab you can find Edit Queries.
2. In the Table Level Options you can find Edit Query.
3. While loading the table edit option that takes you to the Edit Queries.

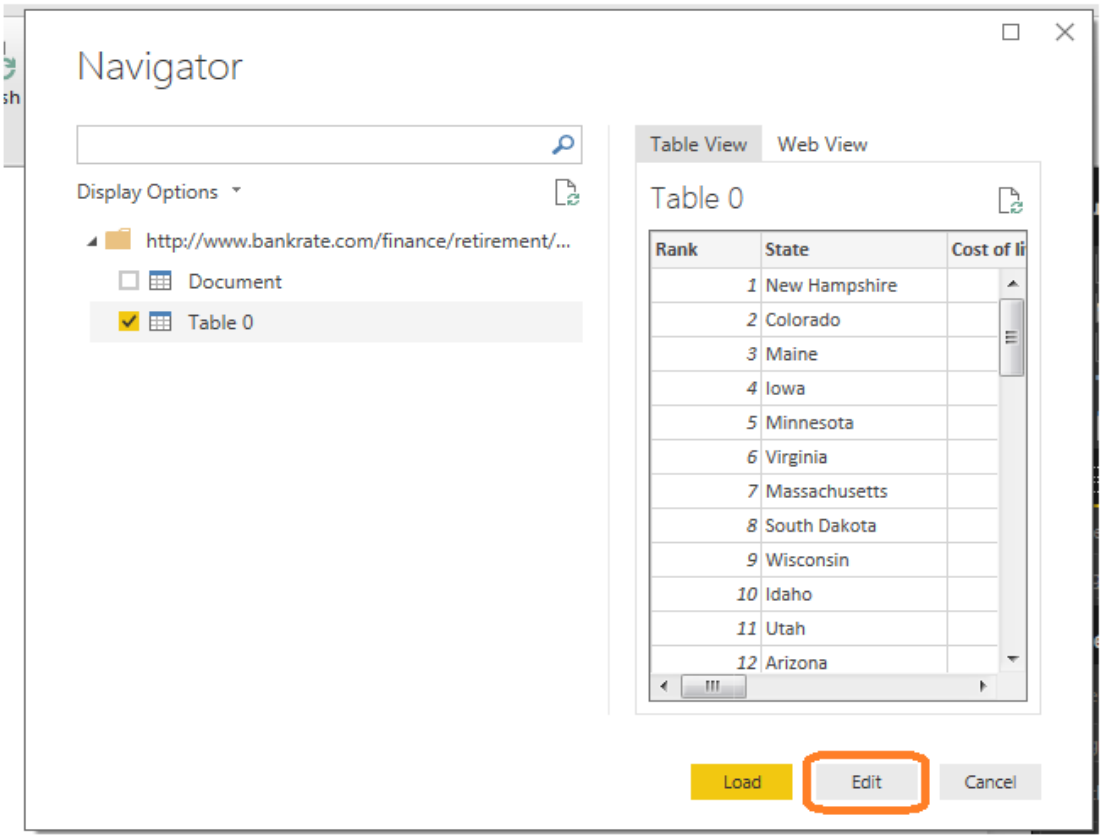
To get into Query Editor, select Edit Queries from the Home tab of Power BI Desktop.



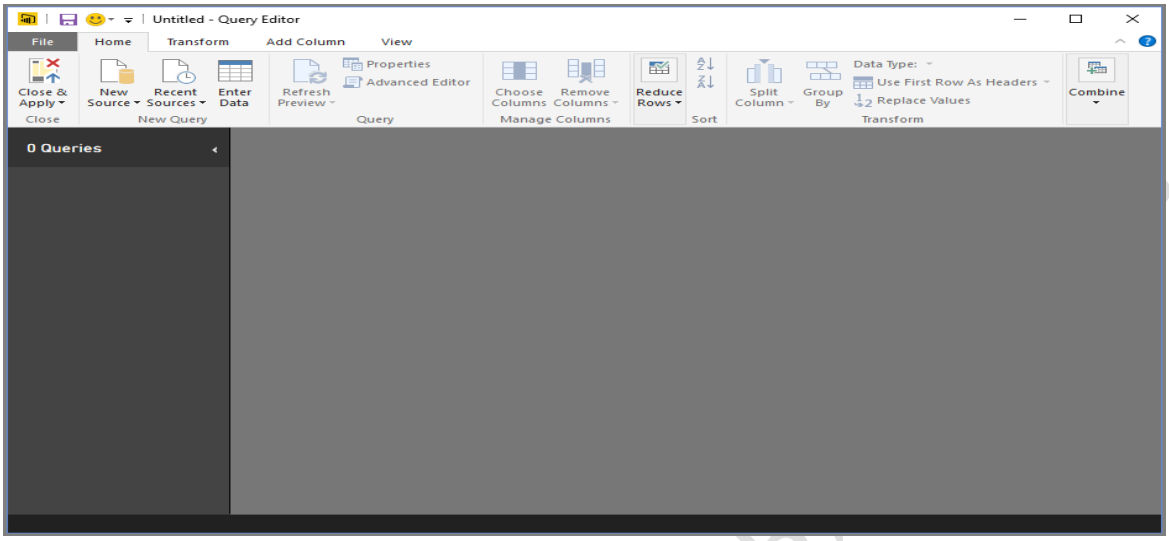
Other way to get or open Query Editor is, go to Table Level Options you can find Edit Query.



Third way is while loading the table "Edit" option that takes you to the Edit Queries.



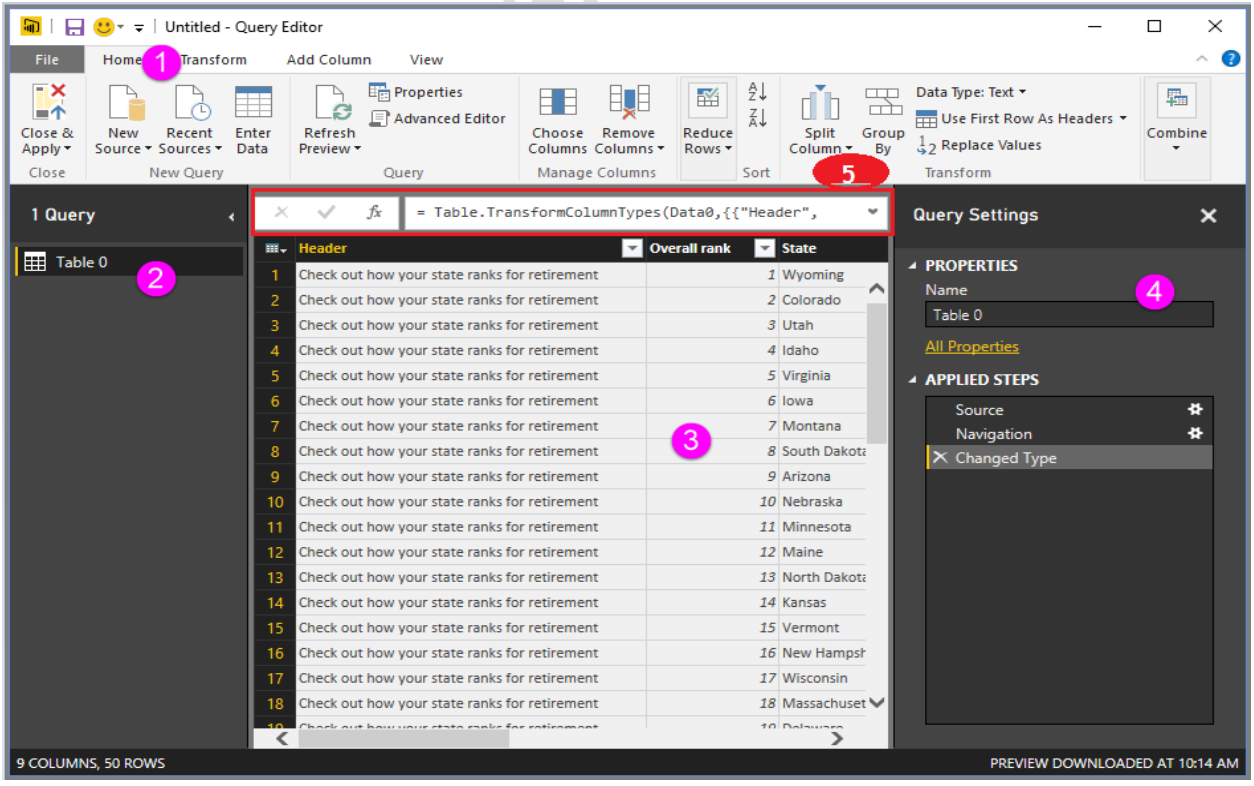
With no data connections, Query Editor appears as a blank pane, ready for data as shown below.



How to establish connection to the source?

Home Tab →New Source → Get Data window→Select the Source Type → Select the Source → Ok

Once Query Editor is loaded with data that's ready for you to shape, you see a handful of sections. Here's how Query Editor appears once a data connection is established.



1. In the ribbon, many buttons are now active to interact with the data in the query for data preparation.
2. In the left pane or queries pane, queries (one for each table, or entity) are listed and available for selection, viewing, and shaping.
3. In the center pane or Results Pane, data from the selected query is displayed and available for shaping.
4. The Query Settings window appears, listing the query's properties and applied steps.
5. The Formula bar is the place where you can see and edit the M code of the current transformation step.

We'll look at each of these four areas

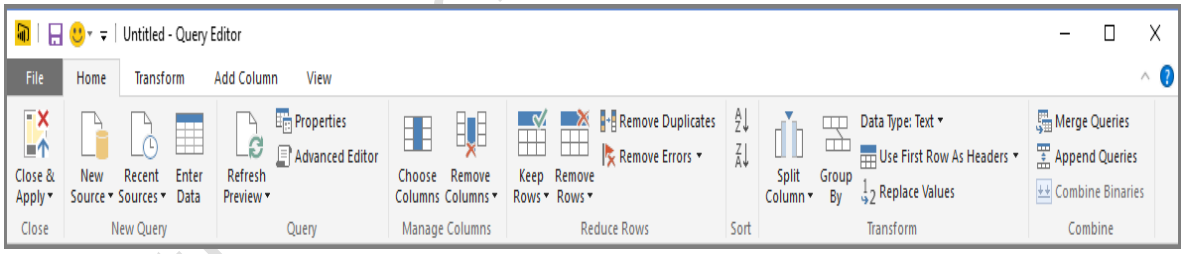
- ✓ The ribbon
- ✓ The queries pane
- ✓ The data view / Results Pane
- ✓ The Query Settings pane
- ✓ Formula Bar

The Query Ribbon

The ribbon in Query Editor consists of four tabs – Home, Transform, Add Column, and View.

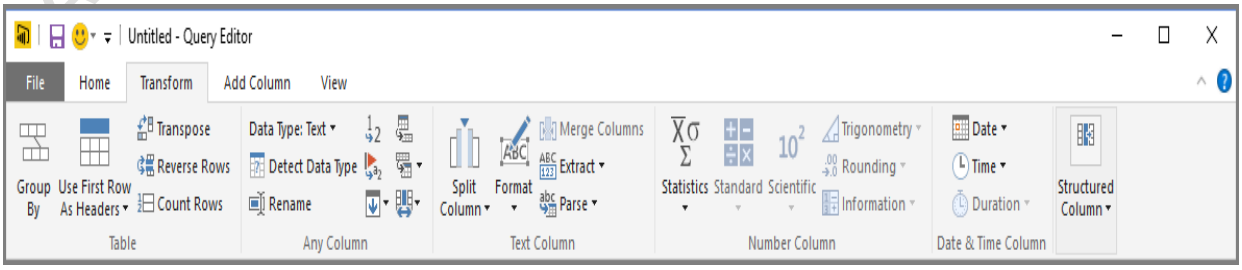
Home Tab

The Home tab contains the **common query tasks**, including the first step in any query, which is Get Data. The following image shows the Home ribbon.



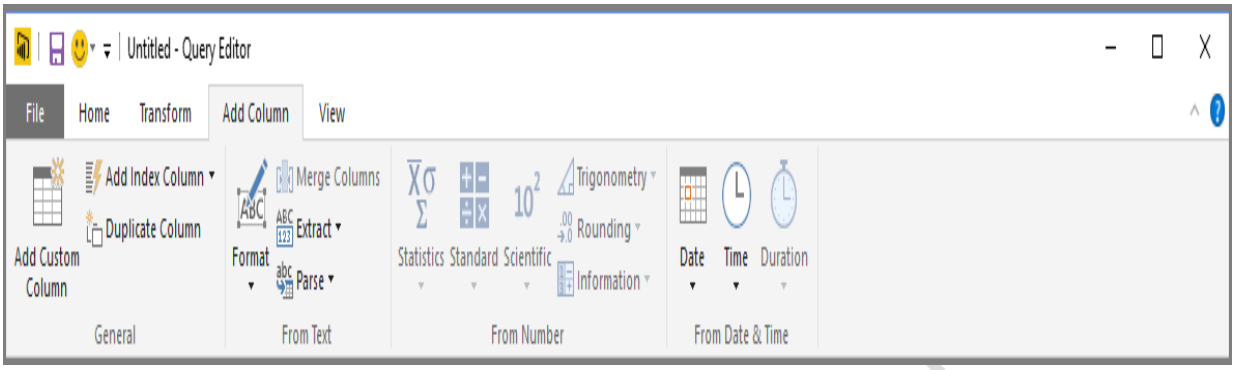
Transform Tab

The Transform tab provides access to common **data transformation tasks**, such as adding or removing columns, changing data types, splitting columns, and other data-driven tasks. The following image shows the Transform tab.



Add Column Tab

The Add Column tab provides additional tasks associated with **adding a column, formatting column data, and adding custom columns.** The following image shows the Add Column tab.



The Difference between the **Transform** and **Add Column** Tabs

The bulk of all transformations available in power query can be accessed through either the Transform tab or the Add Column tab.

You might think there is a lot of duplication between these two tabs. For example, both tabs contain a form Text section with a lot of the same commands. It's not really the case, there is a subtle difference!

When you use a command from the Add Column tab that is found in both tabs, it will create a new column with the transformed data and the original column will stay intact. Whereas using the equivalent command from the Transform tab will change the original column and no new column is created. This is a critical point to be aware of!

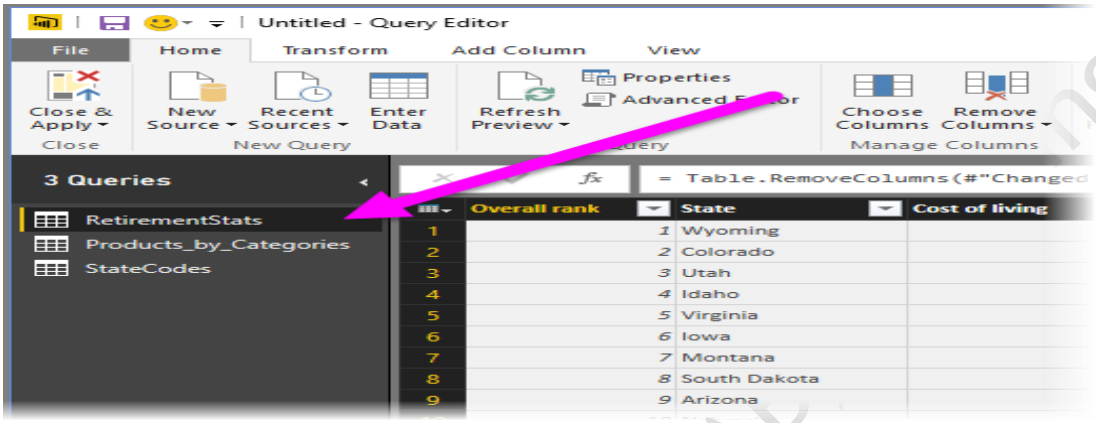
View Tab

The View tab on the ribbon is used to toggle whether certain panes or windows are displayed. It's also used to display the Advanced Editor. The following image shows the View tab.

It's useful to know that many of the tasks available from the ribbon are also available by right-clicking a column, or other data, in the center pane.

The Left Pane / Queries Pane

The left pane displays the number of active queries, as well as the name of the query. When you select a query from the left pane, its data is displayed in the center pane, where you can shape and transform the data to meet your needs. The following image shows the left pane with multiple queries.



The Center (Data) Pane / Results Pane

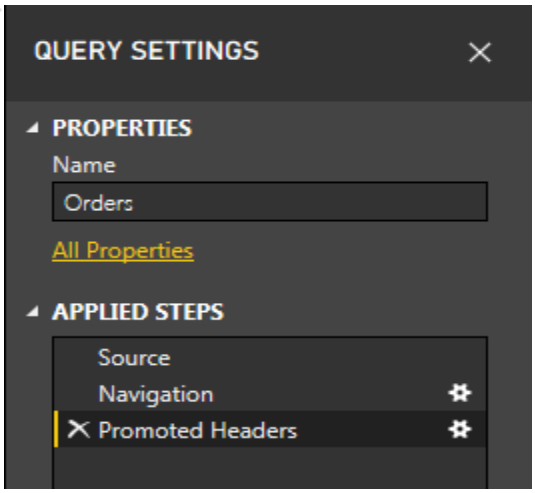
In the center pane, or Data pane, data from the selected query is displayed. This is where much of the work of the Query view is accomplished.

Notice that many of these right-click menu items are the same as buttons in the ribbon tabs.

When you select a right-click menu item (or a ribbon button), Query applies the step to the data, and saves it as part of the query itself. The steps are recorded in the Query Settings pane in sequential order, as described in the next section.

The Query Settings Pane

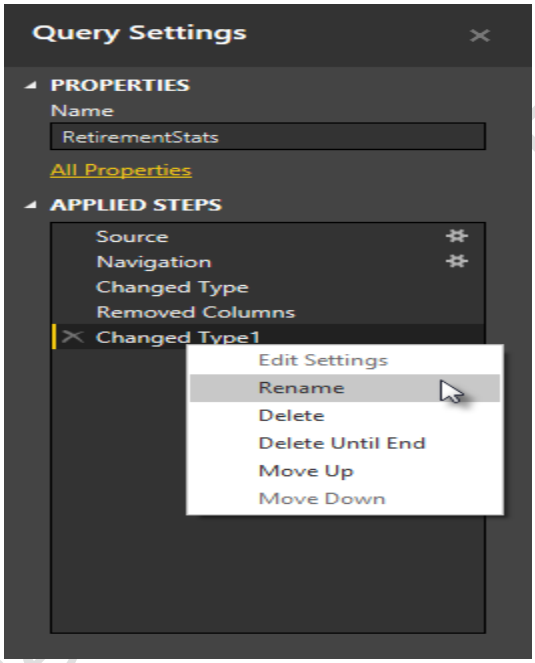
The Query Settings pane is where all steps associated with a query are displayed. For example, in the following image, the Applied Steps section of the Query Settings pane reflects the fact that we just changed the type of the Overall score column.



As additional shaping steps are applied to the query, they are captured in the **Applied Steps** section.

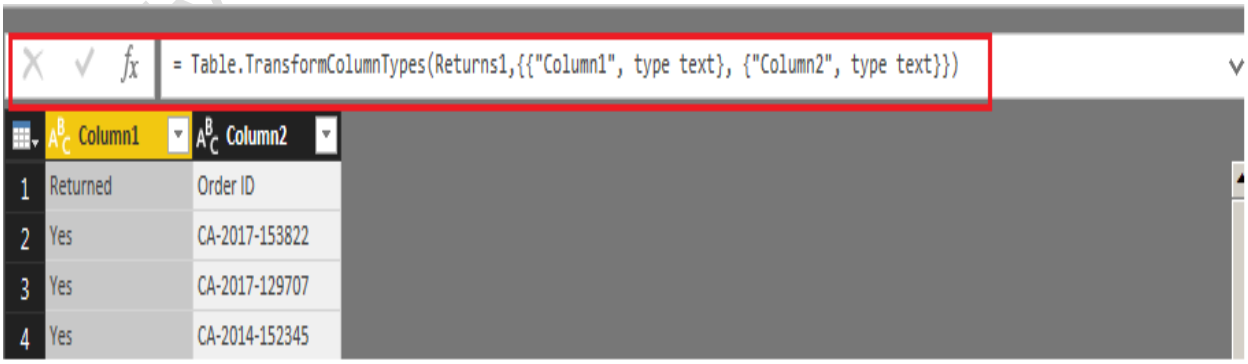
It's important to know that the underlying data is not changed rather Query Editor adjusts and shapes its view of the data, and any interaction with the underlying data occurs based on Query Editor's shaped and modified view of that data.

In the Query Settings pane, you can rename steps, delete steps, or reorder the steps as you see fit. To do so, right-click the step in the Applied Steps section, and choose from the menu that appears. All query steps are carried out in the order they appear in the Applied Steps pane.



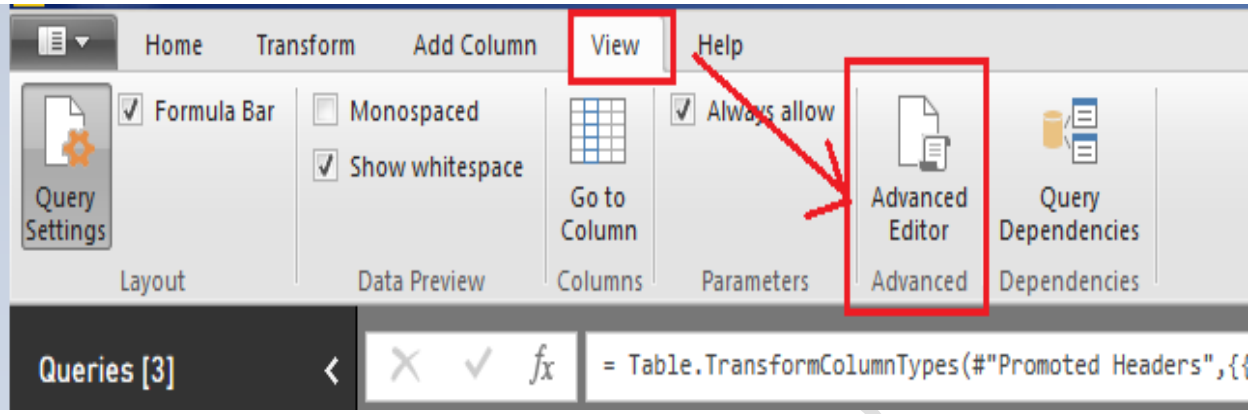
Formula Bar

This is where you can see and edit the M code of the current transformation step. Each transformation you make on your data is recorded and appears as a step in the applied steps area.

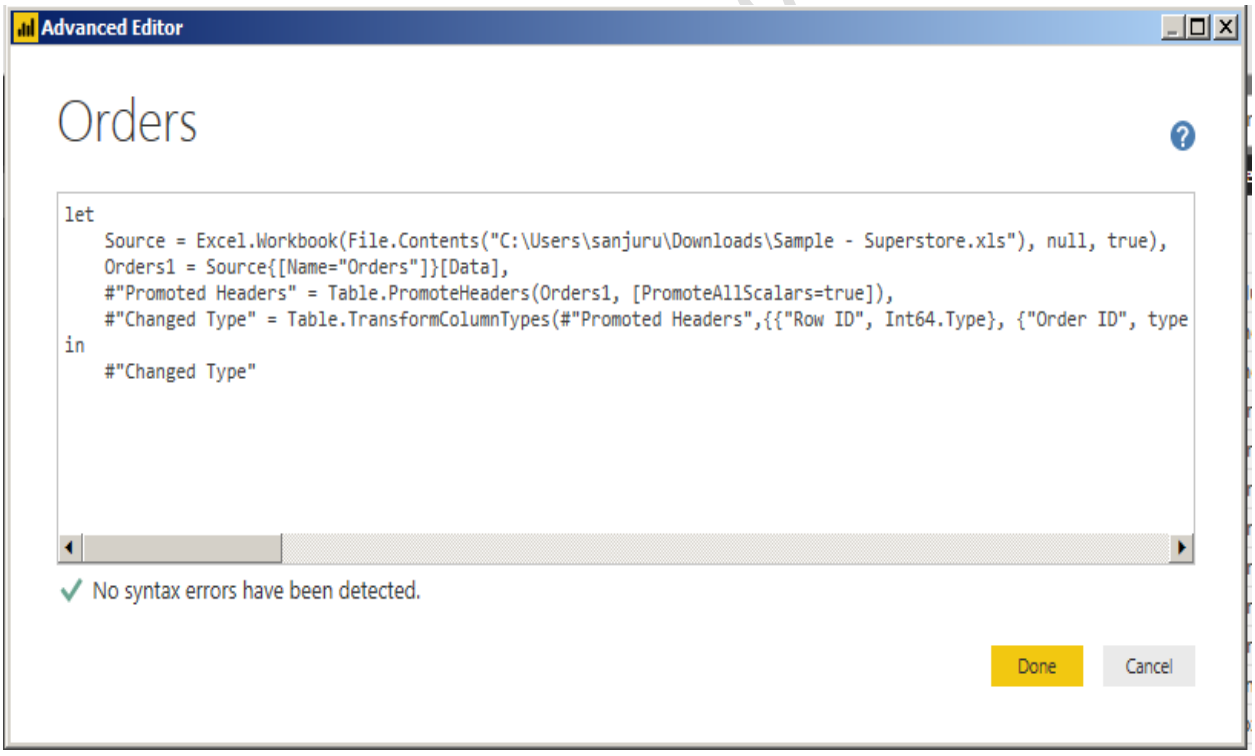


The Advanced Editor

If you want to see the code that Query Editor is creating with each step, or want to create your own shaping code, you can use the Advanced Editor. To launch the advanced editor, select View from the ribbon, then select Advanced Editor.

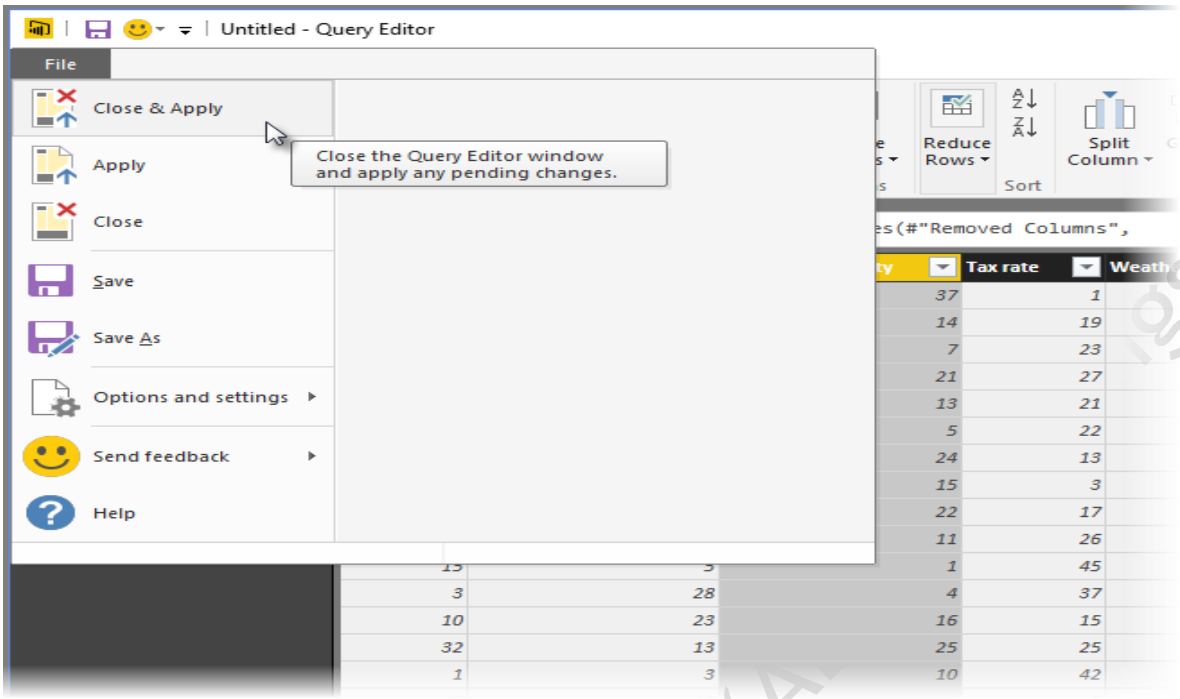


A window appears, showing the existing Query code. You can directly edit the code in the Advanced Editor window. To close the window, select the done or Cancel button.



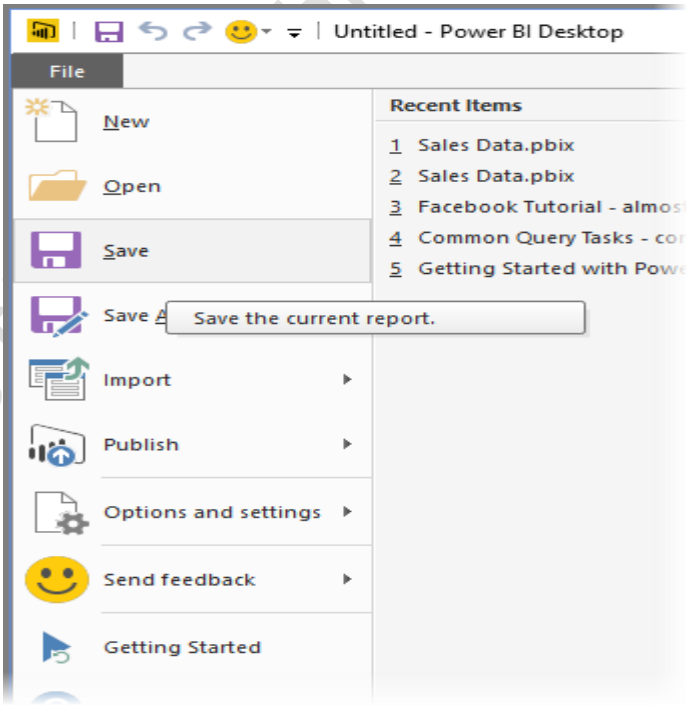
Saving Your Work

When your query is where you want it, you can have Query Editor apply the changes to the data model into Power BI Desktop, and close Query Editor. To do that, select Close & Apply from Query Editor's File menu as shown below.



Once you have your query where you want it, or if you just want to make sure your work is saved, Power BI Desktop can save your work in the form of **". pbix"** file.

To save your work, select File > Save (or File > Save As), as shown in the following image.



In below diagram you can see a high level diagram of Power Query conceptually