

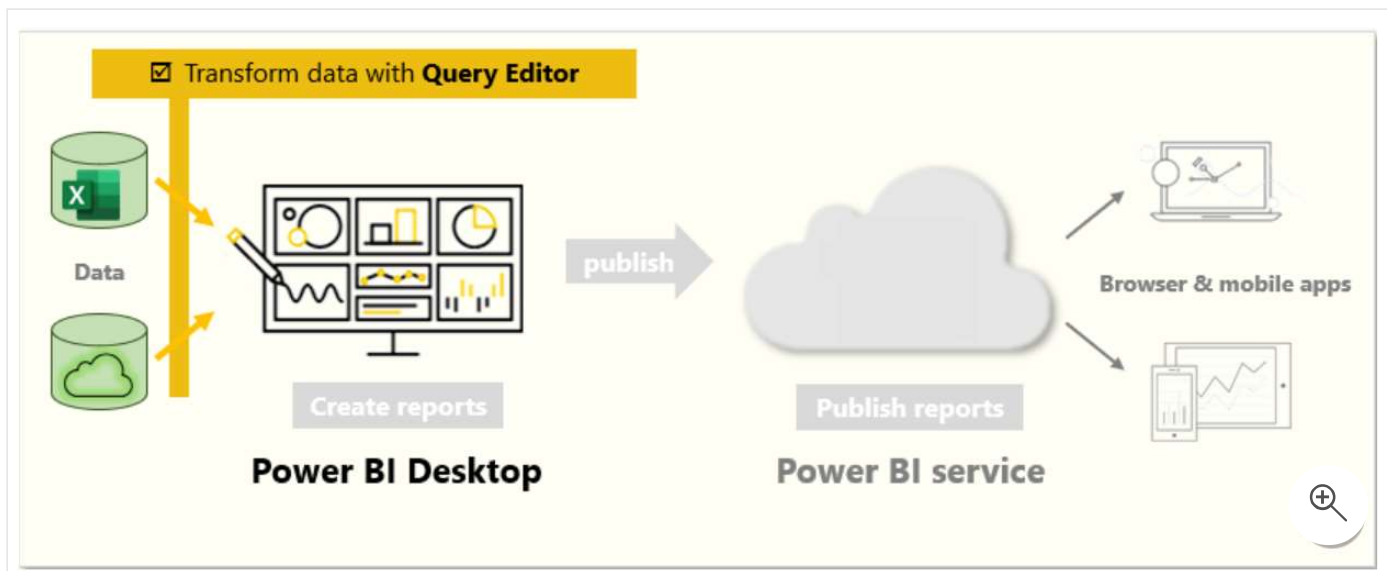
Shape the initial data

18 minutes

Power Query Editor in Power BI Desktop allows you to shape (transform) your imported data. You can accomplish actions such as renaming columns or tables, changing text to numbers, removing rows, setting the first row as headers, and much more. It is important to shape your data to ensure that it meets your needs and is suitable for use in reports.

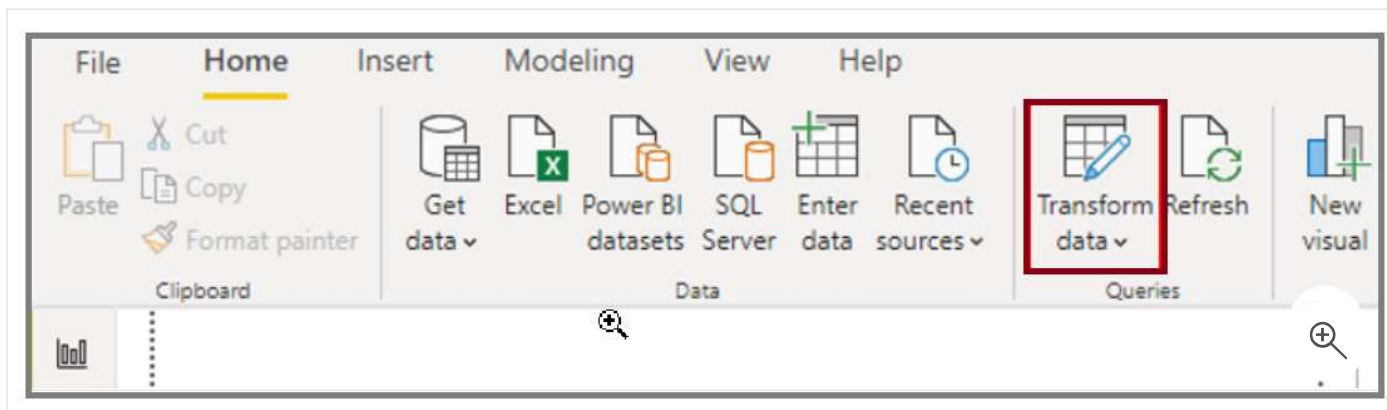
You have loaded raw sales data from two sources into a Power BI model. Some of the data came from a .csv file that was created manually in Microsoft Excel by the Sales team. The other data was loaded through a connection to your organization's Enterprise Resource Planning (ERP) system. Now, when you look at the data in Power BI Desktop, you notice that it's in disarray; some data that you don't need and some data that you do need are in the wrong format.

You need to use Power Query Editor to clean up and shape this data before you can start building reports.



Get started with Power Query Editor

To start shaping your data, open Power Query Editor by selecting the **Transform data** option on the **Home** tab of Power BI Desktop.



In Power Query Editor, the data in your selected query displays in the middle of the screen and, on the left side, the **Queries** pane lists the available queries (tables).

When you work in Power Query Editor, all steps that you take to shape your data are recorded. Then, each time the query connects to the data source, it automatically applies your steps, so your data is always shaped the way that you specified. Power Query Editor only makes changes to a particular view of your data, so you can feel confident about changes that are being made to your original data source. You can see a list of your steps on the right side of the screen, in the **Query Settings** pane, along with the query's properties.

The Power Query Editor ribbon contains many buttons you can use to select, view, and shape your data.

To learn more about the available features and functions, see [The query ribbon](#).

Note

In Power Query Editor, the right-click context menus and **Transform** tab in the ribbon provide many of the same options.

Identify column headers and names

The first step in shaping your initial data is to identify the column headers and names within the data and then evaluate where they are located to ensure that they are in the right place.

In the following screenshot, the source data in the csv file for SalesTarget (sample not provided) had a target categorized by products and a subcategory split by months, both of which are organized into columns.

A	B	C	D	E	F	G	H	I	J	K	L	M	N
ProductSubcategoryID	Name	January	February	March	April	May	June	July	August	September	October	November	December
1	Mountain Bikes	780000	790000	800000	810000	820000	830000	840000	850000	860000	870000	880000	890000
2	Road Bikes	4500	5000	5500	6000	6500	7000	7500	8000	8500	9000	9500	10000
3	Touring Bikes	501000	502000	503000	504000	505000	506000	507000	508000	509000	510000	511000	512000
4	Handlebars	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000	2100	2200
5	Bottom Brackets	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000	2100	2200
6	Brakes	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000	2100	2200
7	Chains	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000	2100	2200
8	Cranksets	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000	2100	2200
9	Derailleurs	1100	1200	1300	1400	1500	1600	1700	1800	1900	2000	2100	2200

However, you notice that the data did not import as expected.

A _C Column1	A _C Column2	A _C Column3	A _C Column4	A _C Column5	A _C Column6	A _C Column7
1		January	February	March	April	May
2						
3	ProductSubcategoryID	Name				
4	1	Mountain Bikes	780000	790000	800000	810000
5	2	Road Bikes	4500	5000	5500	6000
6	3	Touring Bikes	501000	502000	503000	504000
7	4	Handlebars	1100	1200	1300	1400
8	5	Bottom Brackets	1100	1200	1300	1400
9	6	Brakes	1100	1200	1300	1400
10	7	Chains	1100	1200	1300	1400
11	8	Cranksets	1100	1200	1300	1400
12	9	Derailleurs	1100	1200	1300	1400

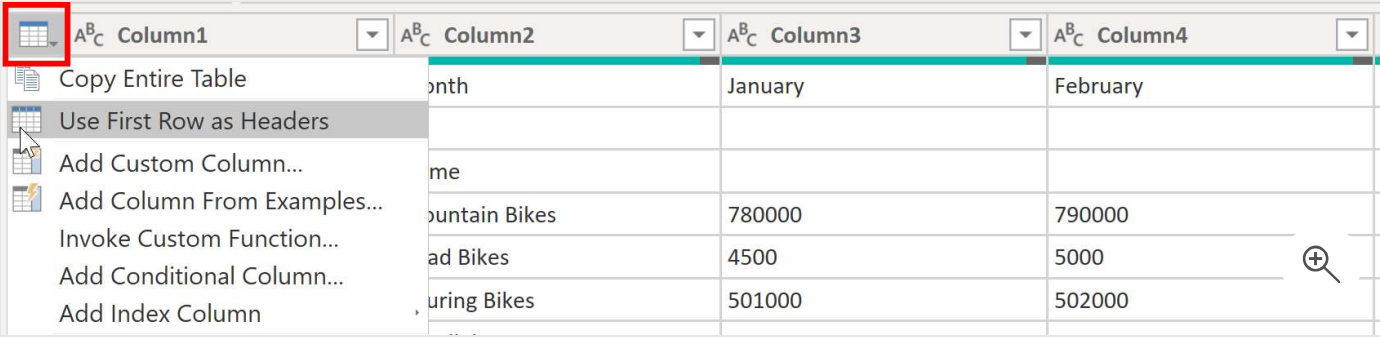
Consequently, the data is difficult to read. A problem has occurred with the data in its current state because column headers are in different rows (marked in red), and several columns have undescriptive names, such as **Column1**, **Column2**, and so on.

When you have identified where the column headers and names are located, you can make changes to reorganize the data.

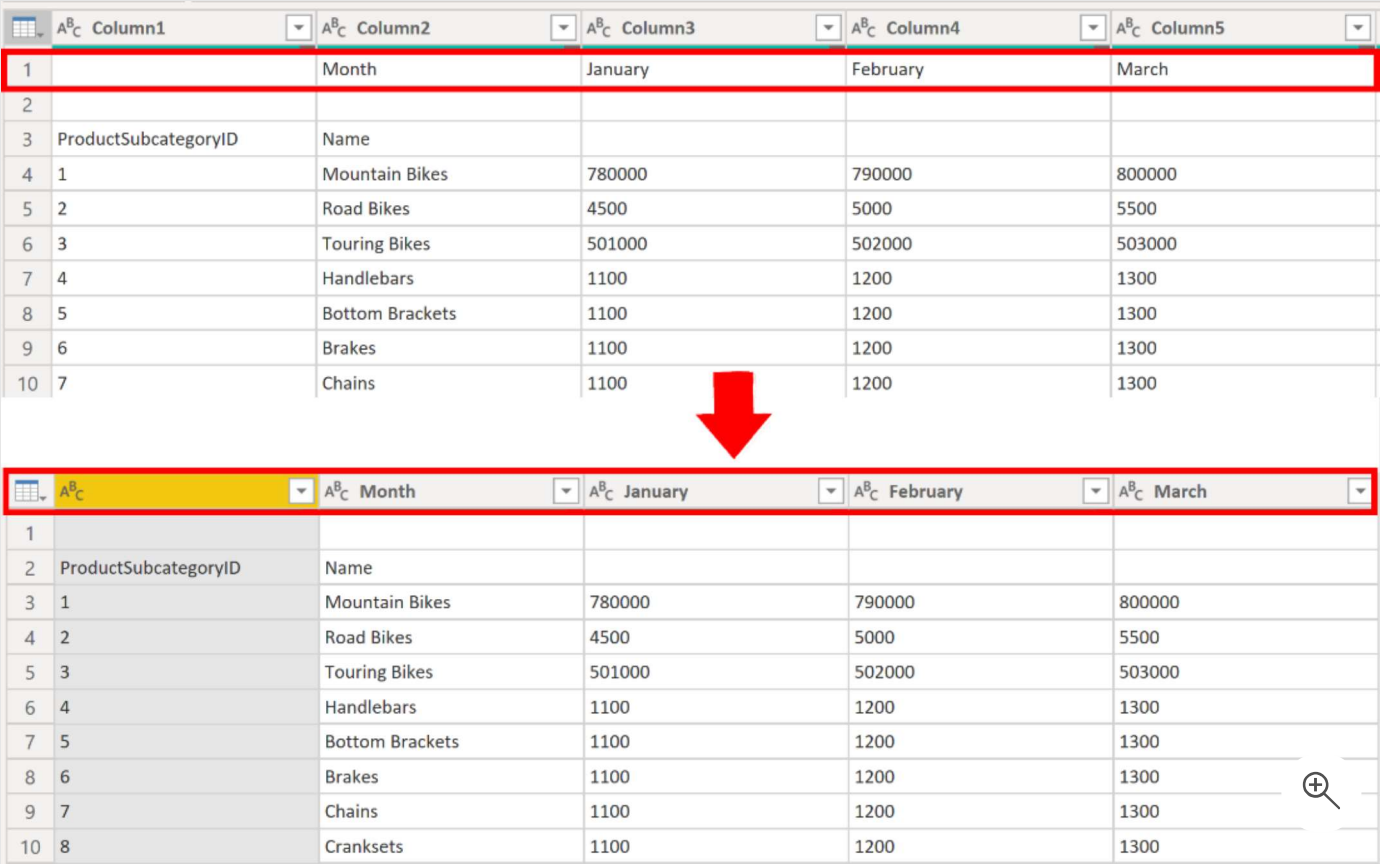
Promote headers

When a table is created in Power BI Desktop, Power Query Editor assumes that all data belongs in table rows. However, a data source might have a first row that contains column names, which is what happened in the previous SalesTarget example. To correct this inaccuracy, you need to promote the first table row into column headers.

You can promote headers in two ways: by selecting the **Use First Row as Headers** option on the **Home** tab or by selecting the drop-down button next to **Column1** and then selecting **Use First Row as Headers**.



The following image illustrates how the **Use First Row as Headers** feature impacts the data:



Rename columns

The next step in shaping your data is to examine the column headers. You might discover that one or more columns have the wrong headers, a header has a spelling error, or the header naming convention is not consistent or user-friendly.

Refer to the previous screenshot, which shows the impact of the **Use First Row as Headers** feature. Notice that the column that contains the subcategory **Name** data now has **Month** as its column header. This column header is incorrect, so it needs to be renamed.

You can rename column headers in two ways. One approach is to right-click the header, select **Rename**, edit the name, and then press **Enter**. Alternatively, you can double-click the column header and overwrite the name with the correct name.

You can also work around this issue by removing (skipping) the first two rows and then renaming the columns to the correct name.

Remove top rows

When shaping your data, you might need to remove some of the top rows, for example, if they are blank or if they contain data that you do not need in your reports.

Continuing with the SalesTarget example, notice that the first row is blank (it has no data) and the second row has data that is no longer required.

	ProductSubcategoryID	Subcategory Name	January	February	March
1					
2	ProductSubcategoryID	Name			
3	1	Mountain Bikes	780000	790000	800000
4	2	Road Bikes	4500	5000	5500
5	3	Touring Bikes	501000	502000	503000
6	4	Handlebars	1100	1200	1300

To remove these excess rows, select **Remove Rows > Remove Top Rows** on the **Home** tab.

The screenshot shows the Power BI ribbon with the 'Home' tab selected. The 'Remove Rows' dropdown menu is open, and the 'Remove Top Rows' option is highlighted with a red box. The ribbon also shows other options like 'New Source', 'Recent Sources', 'Enter Data', 'Data source settings', 'Manage Parameters', 'Refresh Preview', 'Properties', 'Advanced Editor', 'Manage', 'Choose Columns', 'Remove Columns', 'Keep Rows', and 'Remove Rows'. The data table below the ribbon shows the first three rows of the SalesTarget example, with the first row being blank and the second row containing the column headers.

	ProductSubcategoryID	Subcategory Name	January
1			
2	ProductSubcategoryID	Name	
3	1	Mountain Bikes	780000

Remove columns

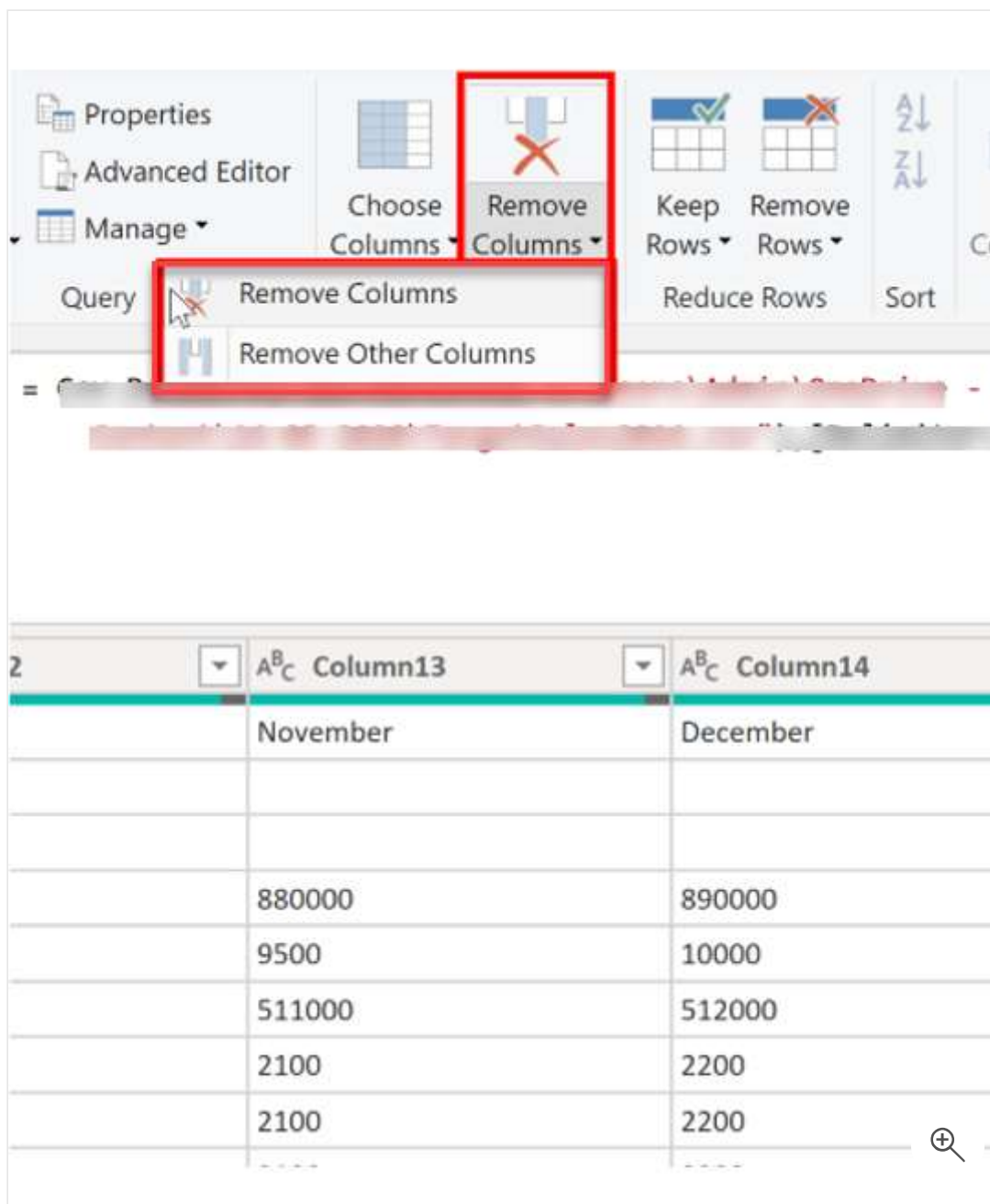
A key step in the data shaping process is to remove unnecessary columns. It is much better to remove columns as early as possible. One way to remove columns would be to limit the column when you get data from data source. For instance, if you are extracting data from a relational

database by using SQL, you would want to limit the column that you extract by using a column list in the SELECT statement.

Removing columns at an early stage in the process rather than later is best, especially when you have established relationships between your tables. Removing unnecessary columns will help you to focus on the data that you need and help improve the overall performance of your Power BI Desktop datasets and reports.

Examine each column and ask yourself if you really need the data that it contains. If you don't plan on using that data in a report, the column adds no value to your data model. Therefore, the column should be removed. You can always add the column later, if your requirements change over time.

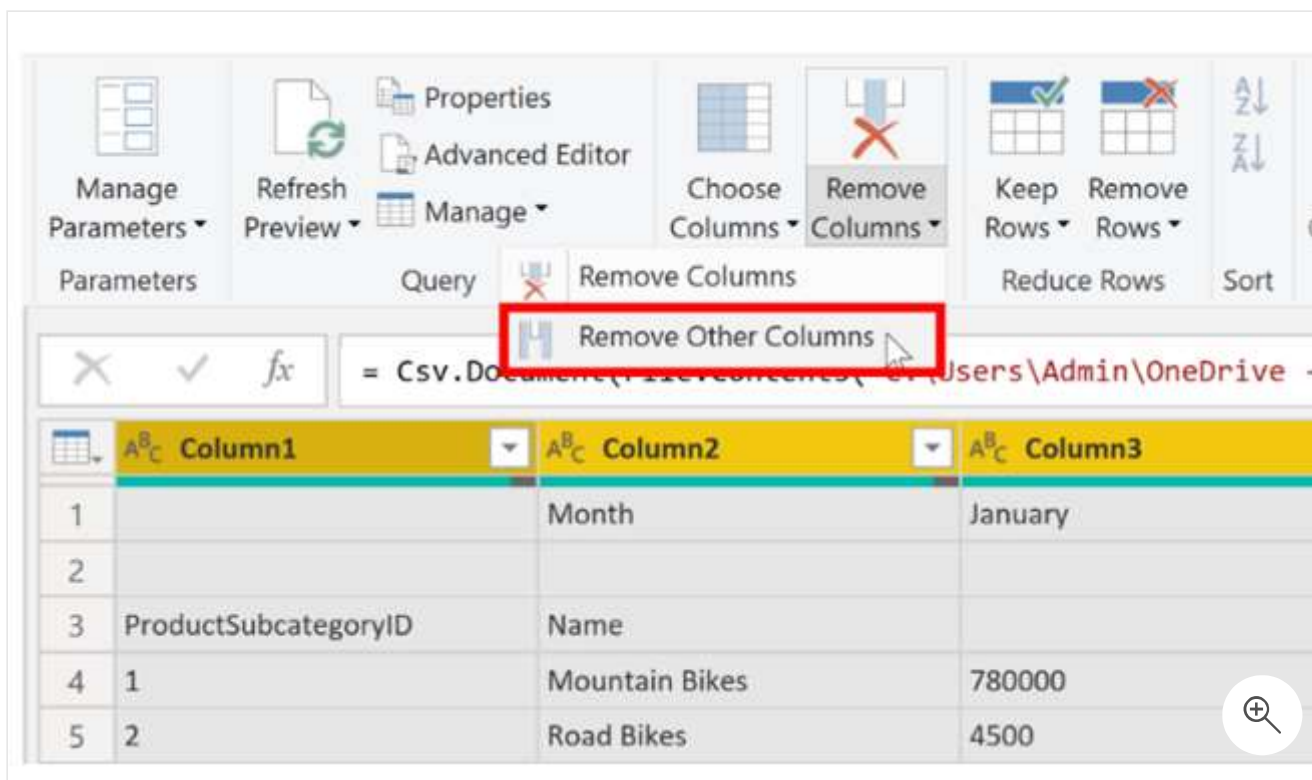
You can remove columns in two ways. The first method is to select the columns that you want to remove and then, on the **Home** tab, select **Remove Columns**.



The screenshot shows the Power BI ribbon with the 'Manage' tab selected. The 'Remove Columns' button is highlighted with a red box. The dropdown menu for 'Remove Columns' is also shown, with 'Remove Columns' and 'Remove Other Columns' options highlighted. Below the ribbon, a table is displayed with two columns: 'A^B_C Column13' and 'A^B_C Column14'. The table contains data for November and December, with values ranging from 2100 to 890000.

	A ^B _C Column13	A ^B _C Column14
	November	December
	880000	890000
	9500	10000
	511000	512000
	2100	2200
	2100	2200
	----	----

Alternatively, you can select the columns that you want to keep and then, on the **Home** tab, select **Remove Columns** > **Remove Other Columns**.



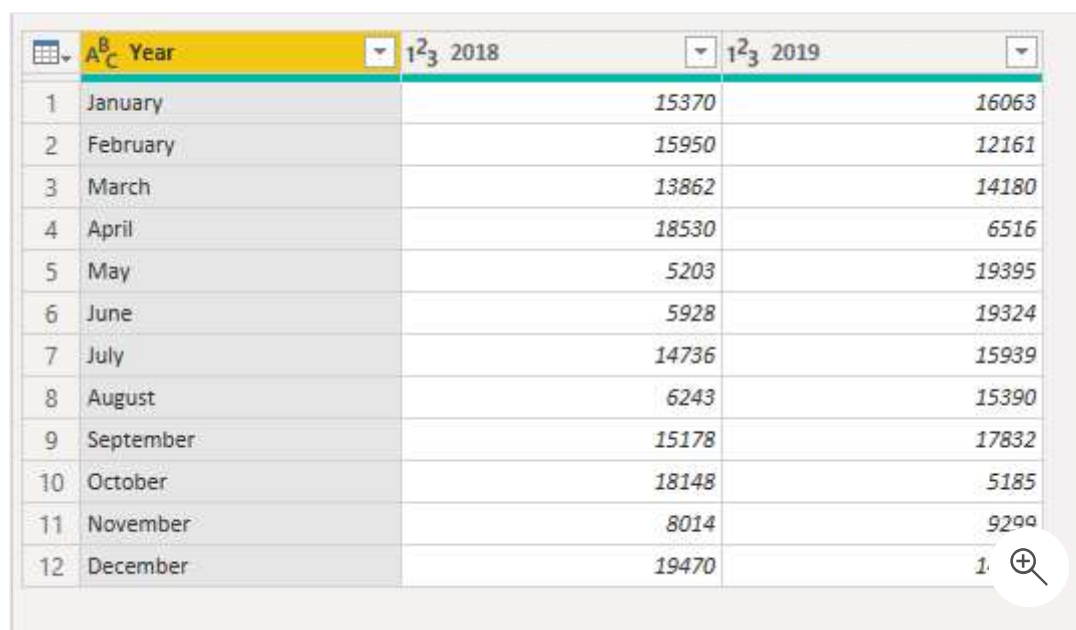
Unpivot columns

Unpivoting is a useful feature of Power BI. You can use this feature with data from any data source, but you would most often use it when importing data from Excel. The following example shows a sample Excel document with sales data.

Year	2018		2019	
January	\$	15,370	\$	16,063
February	\$	15,950	\$	12,161
March	\$	13,862	\$	14,180
April	\$	18,530	\$	6,516
May	\$	5,203	\$	19,395
June	\$	5,928	\$	19,324
July	\$	14,736	\$	15,939
August	\$	6,243	\$	15,390
September	\$	15,178	\$	17,832
October	\$	18,148	\$	5,185
November	\$	8,014	\$	9,222
December	\$	19,470	\$	14,000

Though the data might initially make sense, it would be difficult to create a total of all sales combined from 2018 and 2019. Your goal would then be to use this data in Power BI with three columns: **Month**, **Year**, and **SalesAmount**.

When you import the data into Power Query, it will look like the following image.



The screenshot shows a Power Query table with three columns: 'Year' (containing months), '2018', and '2019'. The rows are numbered 1 through 12. The '2018' column has a filter icon, and the '2019' column has a filter icon and a zoom icon. The data values are as follows:

	Year	2018	2019
1	January	15370	16063
2	February	15950	12161
3	March	13862	14180
4	April	18530	6516
5	May	5203	19395
6	June	5928	19324
7	July	14736	15939
8	August	6243	15390
9	September	15178	17832
10	October	18148	5185
11	November	8014	9299
12	December	19470	1

Next, rename the first column to **Month**. This column was mislabeled because that header in Excel was labeling the 2018 and 2019 columns. Highlight the 2018 and 2019 columns, select the **Transform** tab in Power Query, and then select **Unpivot**.

	^A _C Year	^A _C Attribute	¹ ₂ ₃ Value
	January	2018	15370
	January	2019	16063
	February	2018	15950
	February	2019	12161
	March	2018	13862
	March	2019	14180
	April	2018	18530
	April	2019	6516
	May	2018	5203
	May	2019	19395
	June	2018	5928
	June	2019	19324
	July	2018	14736
	July	2019	15939
	August	2018	6243
	August	2019	15390
	September	2018	15178
	September	2019	17832
	October	2018	18148
	October	2019	5185
	November	2018	8014
	November	2019	9299
	December	2018	19470
	December	2019	14087

You can rename the **Attribute** column to **Year** and the **Value** column to **SalesAmount**.

Unpivoting streamlines the process of creating DAX measures on the data later. By completing this process, you have now created a simpler way of slicing the data with the **Year** and **Month** columns.

Pivot columns

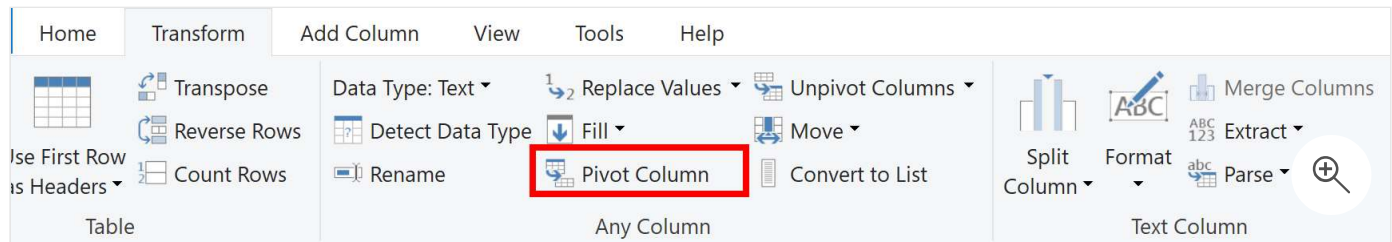
If the data that you are shaping is flat (in other words, it has lot of detail but is not organized or grouped in any way), the lack of structure can complicate your ability to identify patterns in the data.

You can use the **Pivot Column** feature to convert your flat data into a table that contains an aggregate value for each unique value in a column. For example, you might want to use this

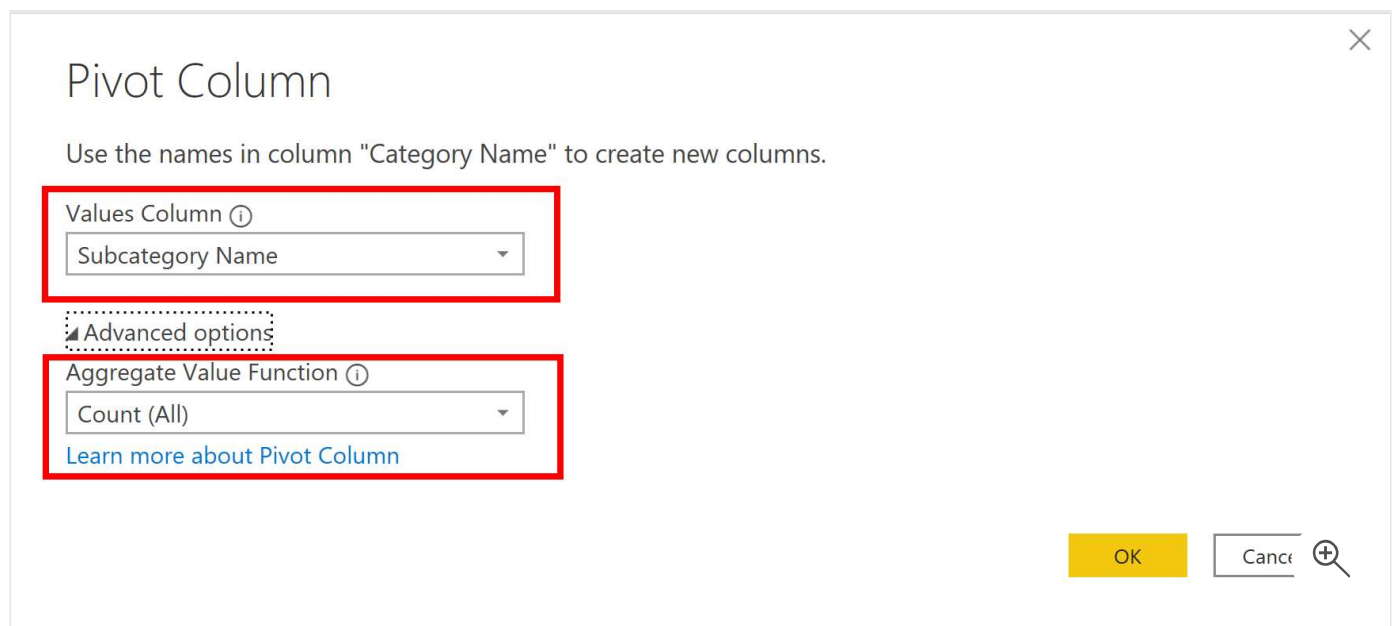
feature to summarize data by using different math functions such as **Count**, **Minimum**, **Maximum**, **Median**, **Average**, or **Sum**.

In the SalesTarget example, you can pivot the columns to get the quantity of product subcategories in each product category.

On the **Transform** tab, select **Transform > Pivot Columns**.



On the **Pivot Column** window that displays, select a column from the **Values Column** list, such as **Subcategory name**. Expand the advanced options and select an option from the **Aggregate Value Function** list, such as **Count (All)**, and then select **OK**.



The following image illustrates how the **Pivot Column** feature changes the way that the data is organized.

	A ^B _C Category Name	A ^B _C Subcategory Name
1	Bikes	Mountain Bikes
2	Bikes	Road Bikes
3	Bikes	Touring Bikes
4	Clothing	Bib-Shorts
5	Clothing	Caps
6	Clothing	Gloves
7	Clothing	Jerseys
8	Clothing	Shorts
9	Clothing	Socks
10	Clothing	Tights
11	Clothing	Vests
12	Accessories	Bike Racks
13	Accessories	Bike Stands
14	Accessories	Bottles and Cages

1.2 Bikes

1.2 Components

1.2 Clothing

1.2 Accessories

1	3	14	8	12
---	---	----	---	----

Power Query Editor records all steps that you take to shape your data, and the list of steps are shown in the **Query Settings** pane. If you have made all the required changes, select **Close & Apply** to close Power Query Editor and apply your changes to your data model. However, before you select **Close & Apply**, you can take further steps to clean up and transform your data in Power Query Editor. These additional steps are covered later in this module.

Next unit: Simplify the data structure

Continue >