Power BI Primer

2022-08-16

Contents

Power BI Primer

What is this?

This book is designed to provide readers with a brief introduction to Power
BI. It is not intended to provide a comprehensive overview of all the features
of Power BI, it should however provide users with enough information to start
transforming and visualising data either to produce an interactive visualisation
or a dashboard.

Contents

- Power BI

    - Introduction
    - Getting Started
    - Data Transformation Basics
    - Data Visualisation
    - Adding New Pages
    - Navigation
    - Indexing
    - Relationships
    - Publishing
    - Sharing
    - Additional Resources

Chapter 1

Introduction

1.1 Introduction

Power BI is part of the **Microsoft Power Platform** which includes other packages such as Power Automate. Power BI is split into several components: Power BI Desktop, Service (an online SaaS - Software as a Service), Gateway, Report Server, Premium, Visuals Marketplace and Mobile Apps. The primary focus of Power BI is business intelligence but Power BI can be used for more than standard reporting of data with line charts and bar charts. It can also be used in conjunction with Python and R to create complex custom visuals or it can be used to create and train machine learning models. Power BI can even be used to web scrape data.

With Power BI Desktop, you can connect to different data sources, transform those data sources through queries, build data models and create visualizations and reports or dashboards for different audiences. Reports and dashboards can be published to the Power BI Service where you can share your reports with other users or a wider audience.

This document is intended to provide a broad overview of the features of Power BI Desktop reports to provide the reader with the necessary tools to start learning and exploring Power BI for themselves. It covers the basics such as importing data, transforming data, visualizing data, using slicer visuals, navigation and basic UI design.

Chapter 2

Getting Started

2.1 Downloading and Installing Power BI

To download Power BI Desktop, go to the Power BI Desktop download page and select **Download Free**. You can also download Power BI Desktop from the Power BI service. Select the **Download** icon in the top menu bar, and then select Power BI Desktop.

2.2 Launching Power BI

Once you launch Power BI Desktop you will be welcomed by the **Start screen**. Here, you can get data sources for your reports, watch video tutorials, check updates and visit the Power BI forums. For now, select the **close** icon to close the Welcome screen to see the main interface (Figure 1 below).

2.3 Important Features

The image above shows some of the most important features in Power BI Desktop.

1. **Get data**: This is used for selecting, connecting to and configuring the data sources that will be used in a report or dashboard. A single data source can be used or multiple can be used. When a data source is selected Power BI will provide you with an option to either load the data or
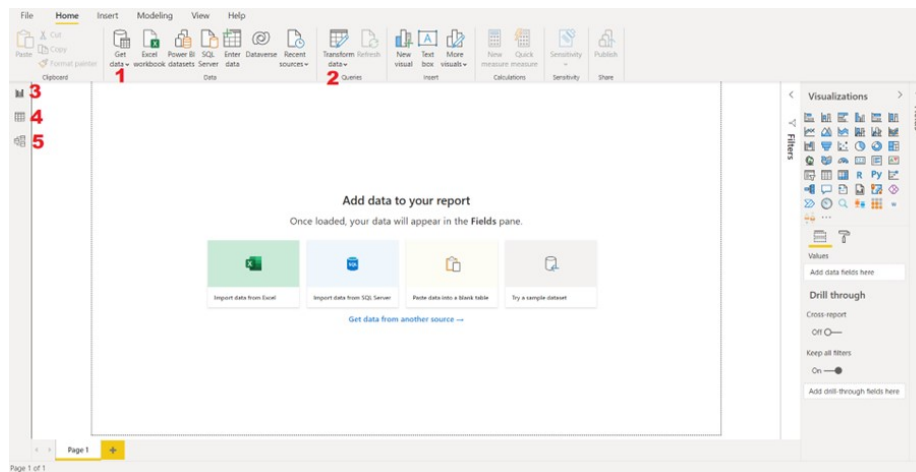
Figure 2.1: Screenshot of the Power BI User Interface.

transform it. It's worth noting that data source will be stored in Power BI's internal memory, so the more data sources you connect to and the larger those sources are the slower your report/dashboard will run.

2. **Transform data**: This is used to launch the Power Query Editor. The Power Query Editor is where data can be transformed. Transformation usually refers to dealing with missing values or cleaning the data before analysis. The Power Query Editor is a useful tool when working with data sets which need to be cleaned. Working with Power Query Editor is similar to working with Excel and often it is useful to switch between the two as some transformations will be easier in one than the other.

3. **Report View or canvas**: The report view icon opens the report view canvas which is used for selecting and designing data visualizations. This is the default view open when Power BI is launched.

4. **Data View**: The data view allows you to view all the data available in your report. This view looks similar to an Excel spreadsheet but it is read only. It is useful to quickly check data types, validate data and make small changes to your data source - such as changing a data type or the format of a variable.

5. **Model View or Relationship View**: The relationship or model view allows you to set relationships between data sources. Relationships occur where two or more data sources are linked together by related data (a shared variable like 'year' for instance).

2.4 Loading Data (Excel)

We're going to use a dummy dataset health_data.xlsx included in the repository used to host this guide. Download and save the dataset in a folder you can easily locate.

To use Excel data sets as a source, open the Power BI Desktop. Under the **Home ribbon** find **Get Data**. Selecting the down arrow next to the Get Data button will show the most common connectors used but clicking the button itself will provide a full list of all available connectors. The list is extensive and Power BI can connect to a wide array of data sources including .csv files, .xlsx and .txt files. Power BI also supports SQL database connections. Whether you click the Get Data button or the arrow you will see Excel at the top of the resulting list. Click **Excel Workbook** and then click Connect. Navigate to your data source.

Once selected you will see two separate spreadsheets you can choose from.
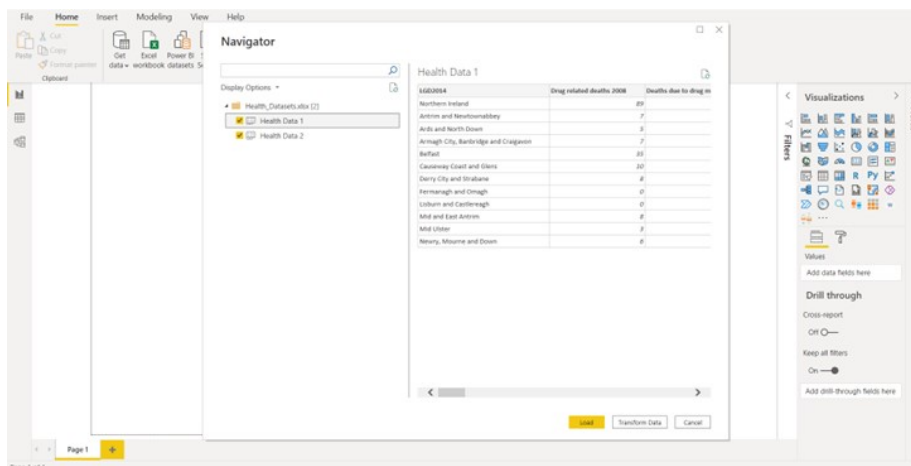
Clicking once on the spreadsheet name will let you preview the data while clicking on the **checkbox** next to the name will include it as part of the data import. Here we have two sheets, Health Data 1 which contains data relating to drug related deaths and deaths due to drug misuse and Health Data 2 which contains population data. This data needs to be cleaned before we can use it so tick both the boxes and then press the **Transform Data** button on the bottom of the panel. This launches the **Power Query Editor**. You can also clean your data in Excel then simply press the Load button but some data sets would be too difficult to transform in Excel.

2.5 Power Query Editor

When the Power Query Editor loads you will notice it has its own environment.
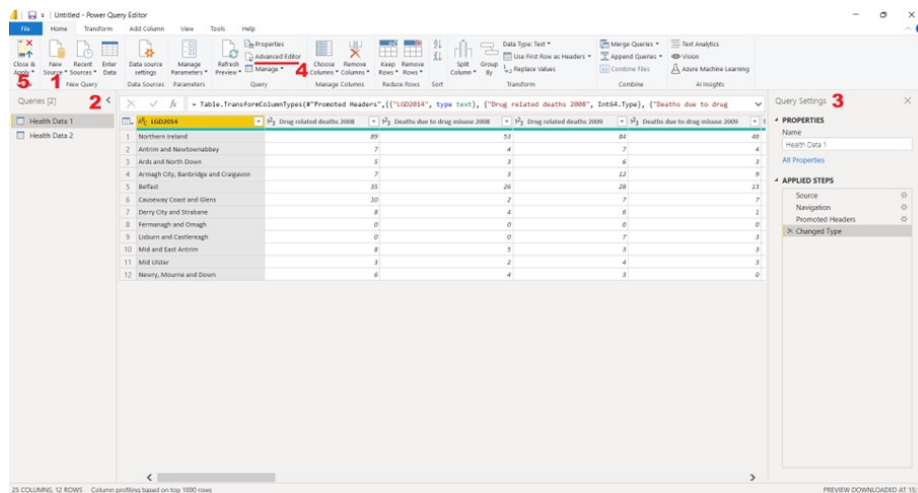
There are a number of features in the Power Query Editor to take note of:

1. **New Source**: This launches the same interface as the Get Data button.

2. **Queries Pane**: The queries pane is a list of all the queries that you have connected to.

3. **Query Settings**: Within this pane you can see the complete history of transformations that have been applied to your query. You can also rename the query. This history is useful for several reasons. From here you can rename a query, or click the 'x' beside a query to remove a query, and reorder queries by clicking and dragging them into different positions.

4. **Advanced Editor**: Power Query Editor uses a programming language informally known as M in the background and when you click any button in the Power Query GUI some M code is generated in the background corresponding to what has been done. By launching the Advanced Editory you can see the M query that is automatically written for you by the Power Query Editor.

5. **Close and Apply**: Choosing this option will save your transformations, close the Power Query Editor and load the transformed data into the data model.

The Power Query Editor allows you to transform the data with step-by-step instructions for adjusting the data. Transforming the data this way **does not affect the original data source**, only this particular view of the data. Data transformation includes a range of processes including: renaming columns or tables, removing rows, replacing missing data entries or changing data types. Power Query Editor records the steps taken to transform the data sequentially under **Applied Steps** in the **Query Settings** pane. If you make a mistake or

transform the data in a way you aren't happy with you can simply cancel the transformation by clicking the **X** that appears beside it in **Applied Steps**.

Chapter 3

Data Transformation Basics

3.1 Data Transformation Basics

In its current format the data contained in Health Data 1 will be hard to visualize in Power BI. We essentially have two data sets in one. The quantitative data is tied up with qualitative data (Drug related deaths 2008). Ideally, we would like the data to look like this:

Table 3.1:

Location	Dataset	Year	Total
Northern Ireland	Drug related deaths	2008	89
Northern Ireland	Deaths due to drug misuse	2008	53
...	...	...	...

Note how the **'year'** data and **'total'** data now have their own columns.

Power Query Editor can help us get there. If we were to unpivot the columns and separate the year into its own column we would arrive at a data set that looks like the table above.

The first step then is to click on the column titled **"Drug related deaths 2008"** then scroll to the right and **shift click** on the final column. This will select all but the first column. Right click on any of the selected columns and select **Unpivot Columns** from the menu. This separates the columns into

attribute-value pairs. Next we need to separate out the year data. **Right click** on the newly generated **Attribute column** and select **Split Column** then select **By Delimiter**.
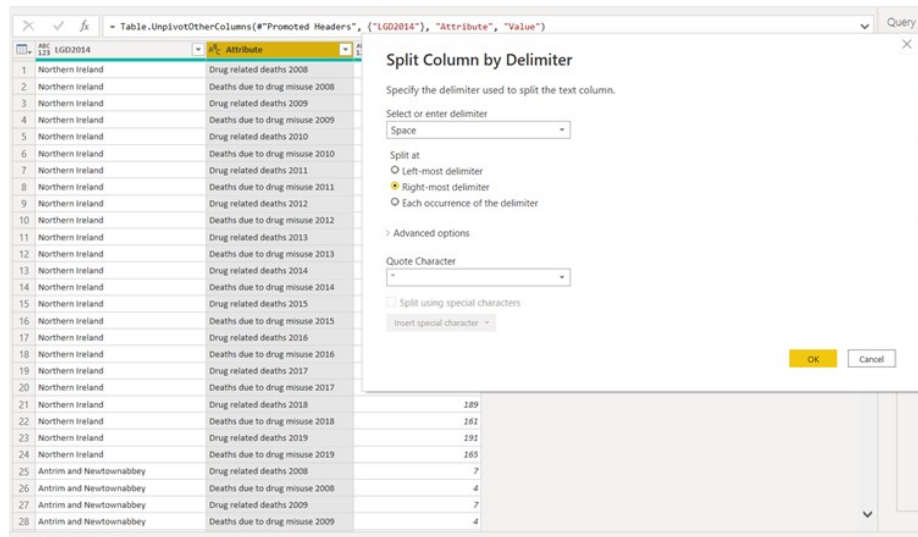

Figure 3.1: Splitting by Delimiters.

We want to split this column into two new columns using a delimiter. The best one to choose here is the Space delimiter. We want to split at the **right-most** delimiter as there is a space right before each year but none after the year. Select these settings and click OK. Change the titles of the columns by **double clicking** the **column header**. Finalize your changes by clicking on the **Home ribbon** and **Close & Apply**. As soon as you click Close & Apply in the Power Query Editor the Power Query Editor will close and the Power BI Desktop Report View will be visible.

These are two of the most common data transformations that data sets will need to have applied to them.

Chapter 4

Data Visualisation

4.1 Visualising Data

Power BI is commonly associated with the ability to create impactful, interactive data visualizations. Before we start making visuals it's worth detailing some of the panes and views available to us:
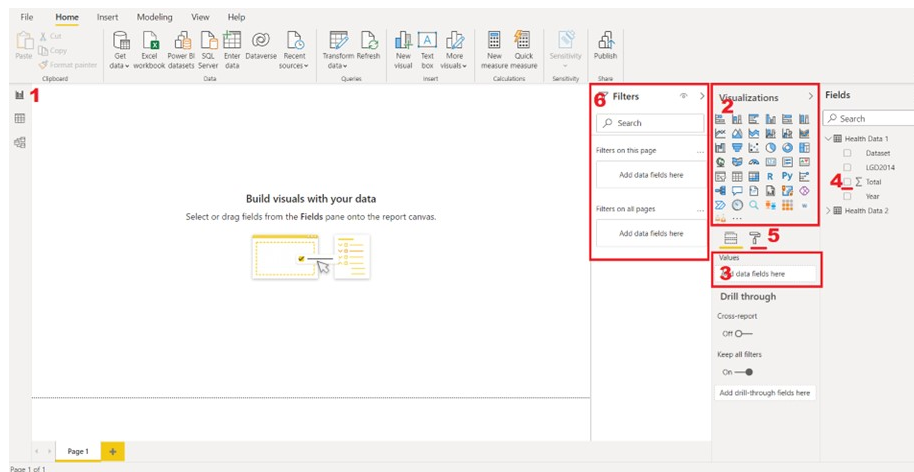

Figure 4.1: Navigating the user interface.

1. **The Report View**: This is the button that will open up the Report canvas and allow us to create visuals using the visuals area. Below the

17

report view button are two more buttons. These provide access to the
data view and the **model view**.

2. **Visuals Area**: This is where we can choose which visual we would like
   to use. Each of the icons represents a type of visualization. Clicking these
   icons will generate a visual that we can start to work with. Once custom
   visuals are added they will appear here as well.

3. **Field Area**: This area will change depending on the visual but it is
   where we place the variables (Total, Year... etc.) that we will use within
   the selected visual.

4. **Field Pane**: This pane contains the fields we have to choose from to add
   to our visual. Initially this is populated by the variables contained in data
   sources but it is possible to add new variables or measures through the
   Power Query Editor.

5. **Format Area**: This is where we can decide on the formatting of our
   visuals. You can customise colours or make changes to axes add borders
   and change font sizes... etc.

6. **Filters Area**: This is where we can apply filters of various scopes. We
   can apply page-level filters which will affect all the visuals on the selected
   page. We can apply Report-level filters which will affect every visual in
   the entire Power BI Report and we can apply Visual-level filters which
   will be applied only to specific visualizations.

4.1.1   Creating a Line-Chart

We're going to create a **line-chart visual** that shows both drug related deaths
and deaths due to drug misuse. We want to show the totals for each over time
and we want to be able to filter by local government district.

The first step in creating any visual is to go to the **Visualizations pane** on
the right-hand side of the UI. Here you can see a range of different visualization
icons representing their respective chart. Along the top row are bar charts, in
the second row are line charts, area charts and a ribbon chart, below these are
pie charts, maps, cards, and even tools to create custom Python and R visuals
(Plotly, Matplotlib and Seaborn are three commonly used Python plotting li-
braries). Find the Icon for the **Line Chart** and click it. A new visual object
will appear on the canvas with a prompt to select or drag fields to populate this
visual.

Drag the **"Year" field** from the **fields pane** into the **Axis box** in the field
area. Drag the **"Total" field** into the **Values box**.
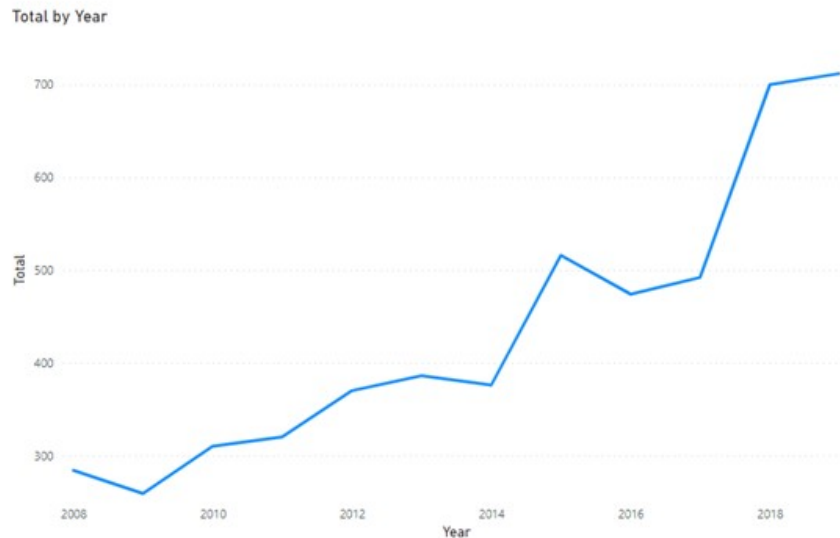
You should get a line-chart that looks like this:

Figure 4.2: The first line chart.

There is something not quite right about this visual. We plotted Total by Year but what is it a total of? Our data contained totals for each Local Government District and for two different data sets. Currently, the data for each year is being summed. We are looking at a line chart showing the sum of all Local Government District drug related deaths and deaths due to drug misuse. We can tell this is the case at a glance by looking to the **Fields Pane** where the **Total** has a summation sign ($\sum$) to the left of it.

Drag the **Dataset field** into the **Legend box** to separate out the data sets. You should see a line-chart like this:

This is closer to what we had hoped to show but it's still summing over the local government districts. This is where the **Slicer visualization** becomes useful. Although, we don't need to use slicers, we could also use the filters pane to simply remove the districts we don't want to be summed. This can be useful when datasets contain rows of data that we aren't interested in.

4.1.2 The Slicer Visual

The **Slicer visual** is used when we want our users to be able to filter by something that isn't used inside our visuals. The slicer visual only allows one field to be assigned to it. Passing text data into the field will generate a list of check boxes which can be ticked or unticked. Passing numeric data into the field will
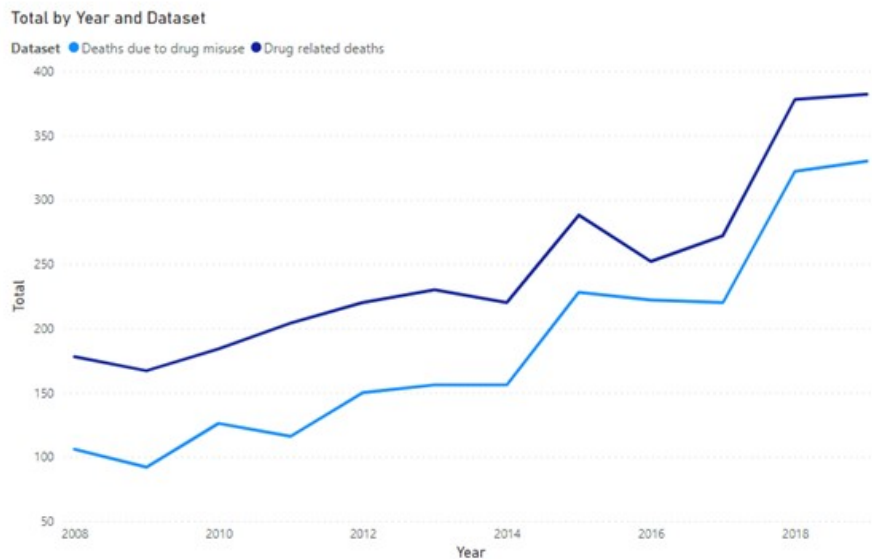
Figure 4.3: Line chart.

generate a slider which can be moved from one numeric value to the next (useful for dates).

Before we create a new Slicer visual it is important to make sure we have deselected the line-chart visual otherwise clicking on a new visual icon will convert this visual into the selected visual type. We don't want to convert our visual, we want an entirely new visual. **Click** on **the canvas** behind the line-chart visual then in the **visualizations pane** locate the **Slicer visual** and click on it. A new visual object will appear. Drag **LGD2014** from the **Fields Pane** into the **Field box** in the Field Area. A checkbox list of local government districts should populate in the visual object on the canvas. It should look like this:

4.1.3   The Format Area

We now have a line-chart that shows both deaths due to drug misuse and drug related deaths but if we control click multiple local government districts it sums them together. We can change this behaviour by formatting the slicer visual.

Click the **slicer visual** then click on the **Format area selector** (the icon that looks like a paint roller). We will be presented with a list of different attributes that we can change. We can change the title, the background, the border and we can add shadows. The important attribute we want to change however is **Selection controls**. Click the **drop-down arrow** beside **Selection controls**
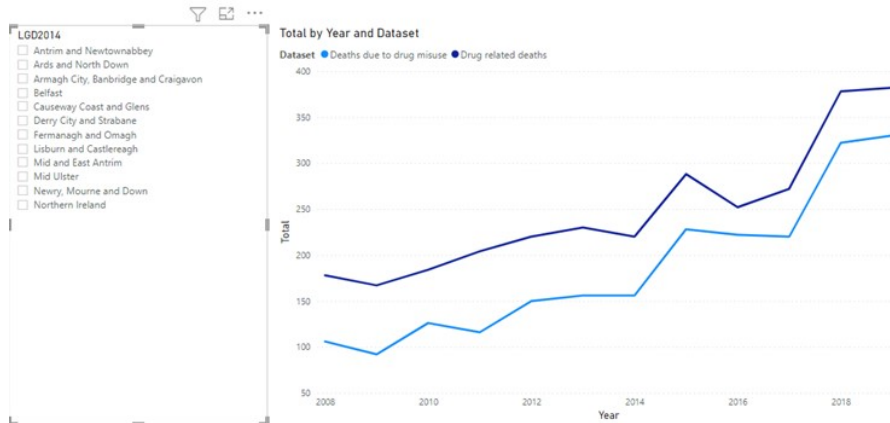
Figure 4.4: Filtering data by local government district (LGD).

and toggle **single select** on. The result should be something that looks like this:

This presentation style has positives and negatives. It's useful if we know we have lots of visitors who only want to know about their local government district but it's not very good for comparing different districts. We're going to use what we have done so far to build a new visual that allows users to directly compare their local government district to another.

Our line-chart visual will use LGD2014 as its legend. Our slicer will have to enable multiple selection. We may also want to include additional slicers to filter the visual by year and by the data set we would like to compare. We will also convert the visual type to a stacked bar chart. The final result should look like this:

The new visual should allow us to check multiple boxes and add to the stacked bar chart. We should also be able to filter by year using the slider at the top and we should be able to select the data set we're interested in using the buttons above the visual.
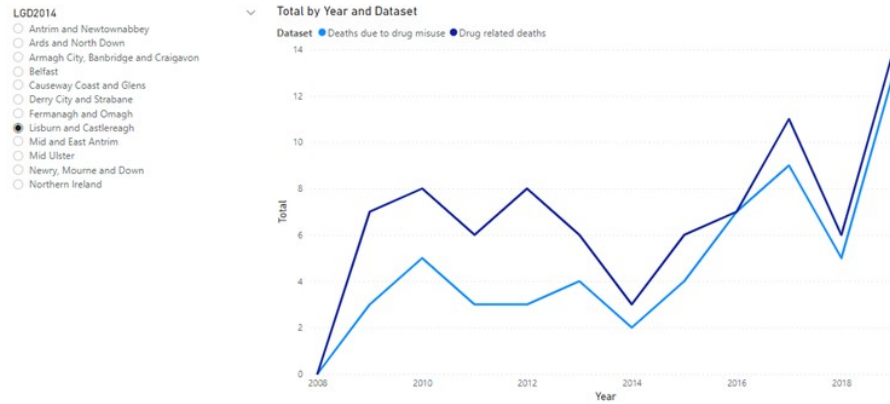
Figure 4.5: Filtering data by local government district (LGD) and selecting different regions.
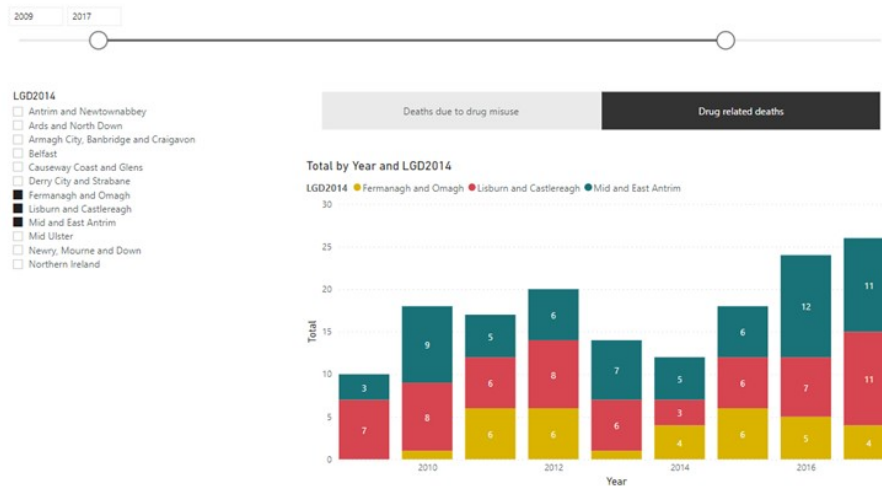

Figure 4.6: The Finished Product.

Chapter 5

Adding New Pages

5.1   Adding a New Page

We can add a new page to our report by clicking the plus sign $+$ next to the **Page 1 tab** at the bottom of the canvas. It makes sense however to duplicate our existing page since we can re-use some of our existing work. **Right click** on the **Page 1 tab** and select **Duplicate Page**.

Select the **line-chart visual** and change the **legend** in the **Fields Area** from **Dataset** to **LGD2014** by clicking on the **X** to the right of **Dataset** and dragging in **LGD2014** from the **Fields Pane**. Now click on the **stacked column chart** icon in the **Visualizations Pane** to convert the visual. Some conversions work better than others, it's always worth checking that the visual is showing the same data before and after conversion. A conversion from a stacked bar chart to a card visual or KPI visual may not be the best choice for instance.

We will need to edit the location filter (the slicer) to allow for multiple selections. Click on the **slicer visual** and go to the **Format Area**. Under **Selection Controls** toggle **Single Select** to **off** and ensure that **Multi-select with CTRL** is toggled to the **on** position and **Show "Select All" Option** is toggled to the **off** position. We can now select multiple local government districts and they should stack on top of one another in the stacked bar chart visual. We will need two new slicer visuals.

Click on the canvas to ensure no visuals are selected. Select the **Slicer Visual** in the visualizations pane. Drag the **Year field** into the **Field box**. The slicer should automatically detect numeric data and generate a slider that takes values in the range 2008 to 2019.

We need one more slicer. Again ensure no visuals are currently selected and select the **Slicer Visual**. Drag the **Dataset** field into the slicer's field. A checkbox will be generated. We would prefer buttons this time, so go to the **Format Area** and select the **General** option. Scroll down until you see **Orientation**. Click on the drop down menu and select **Horizontal**. Under **Selection Controls** you might want to toggle **Single select** to **on**.

Finally, we can add some data labels to our visual. Select the line-chart visual by clicking on it. In the Format Area scroll down until you see **Data Labels** and toggle them **on**.

Chapter 6

Navigation

6.1 Navigation

There are a number of methods to provide users with a means to navigate their way through a dashboard.

6.1.1 Arrows

Up to this point we have mostly made use of the Home ribbon but there are other important ribbons when designing reports. The **Insert ribbon** for instance allows us to insert elements into our report that aren't necessarily data visualizations. We can insert text boxes, buttons, shapes and images. These are all useful for providing additional information on visualizations, explaining data or navigation. We're going to create navigation buttons.

Add two new pages to your report using the plus icon next to the page tabs at the bottom of the canvas. Click onto the first of those pages. Then navigate to the **Insert ribbon** and click on **Buttons** then on the **right arrow**.

Repeat this process on the second page but this time select the **Left arrow**. If the Left arrow is selected you will see the visualisations pane disappear and the entire pane will be taken up by the format pane. Scroll down until you find **Action**. Toggle **Action** to **on** and select **Page navigation** in the **Type menu** and **Page 1** (or whatever page you want this arrow to direct to) in the **Destination menu**.

If you **CTRL+Click** on the **Left arrow** it should now take you to **Page 1**. When your Power BI Report is published users won't need to CTRL+Click it
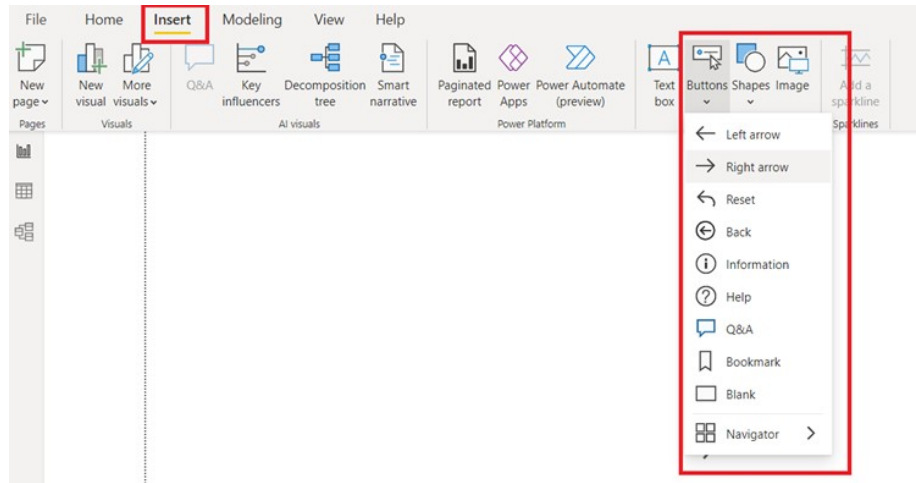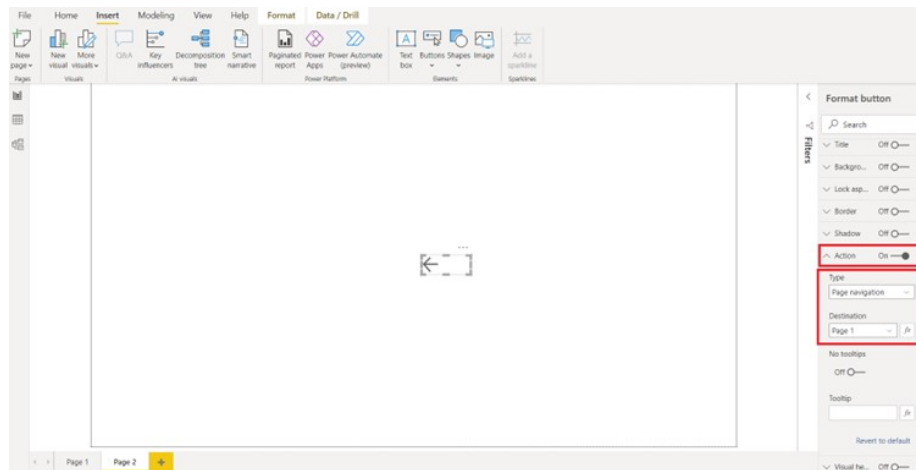
Figure 6.1: Inserting arrows.

Figure 6.2: Left Arrow Navigation.

will just happen with a click. We can repeat these steps for the Right Arrow on Page 1 to cycle between the pages.

6.1.2 Buttons

Another useful element that we can insert is the **Button**. The Button can serve just about any purpose. It can be filled with color and used as a banner or a footer; it can be used as a navigation tool thanks to the actions we just used; and it can be used as a display behind a visual.

Create a new page. Find **Buttons** on the **Insert ribbon**. Click it and select **Blank**. An outlined box should appear. You can click the edges of this to resize it. Try to take up the whole width available and just less than 20% of the height. It doesn't need to be exact, just resize to whatever looks natural as a header for a title. Click on the resized button when you're finished. In the **Format Button pane** toggle **Outline** to **off** and toggle **Icon** to **off**. Toggle **Fill** to **on** and expand the fill menu and select a new color for your header. Change the transparency settings if you like. Next, find the **Text toggle** and toggle it **on** if it isn't already. Open the **Text menu** by clicking the drop-down arrow. In the text box type a title for your new page. You can change the font color and text size and the font below the text entry box.

Add another button but this time resize it to whatever size you think appropriate for a navigation button. Repeat the steps above to add colour and text to the button. Then you can use the action menu, like we did when we created the left and right arrows, to give the button navigation properties.

With any luck you should be able to replicate the kind of user interface you see in a **single page web app** or a **mobile app**.
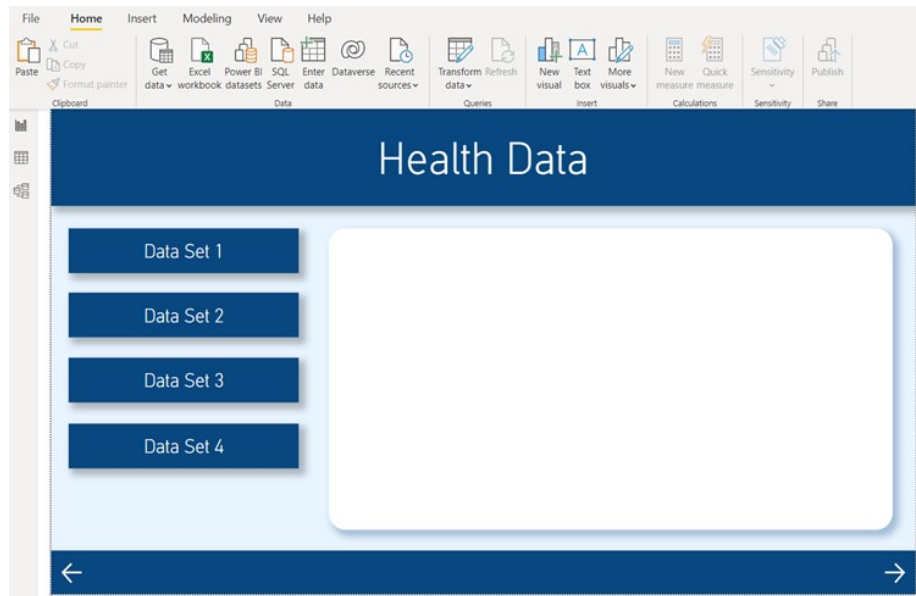
Figure 6.3: Buttons used to create links between pages.

Chapter 7

Indexing

7.1 Indexing

This example will show you how to create a bar chart visual of sales by month. This sounds simple but often when trying to visualise data categorised by month the bars for each month are sorted by alphabetical order rather than the order in which the months occur.

Start a new project and click **Enter Data**.

Enter the months of the year in the available column and change the column title to "Month".

Press the **+** Icon at the top right to create a new column, title it "Sales" and enter some dummy sales data.

Load the data.

Create a bar chart visualisation with "Month" in the X-axis field and "Sales" in the Y-axis field.

Once the visualisation has been created click the **...** icon for more options and sort the axis by month. The x-axis will be sorted by month but it will be sorted alphabetically.

To rectify this we need to create an **index**.

Click on **Transform Data** and navigate to "Monthly Sales". In **Applied Steps** pane on the right hand side there will be a step called **Source** with a cog icon to the right of it. Click on the cog. This will open up the editor you used to input the data. Create a new column by clicking the **+** icon. Call this new column
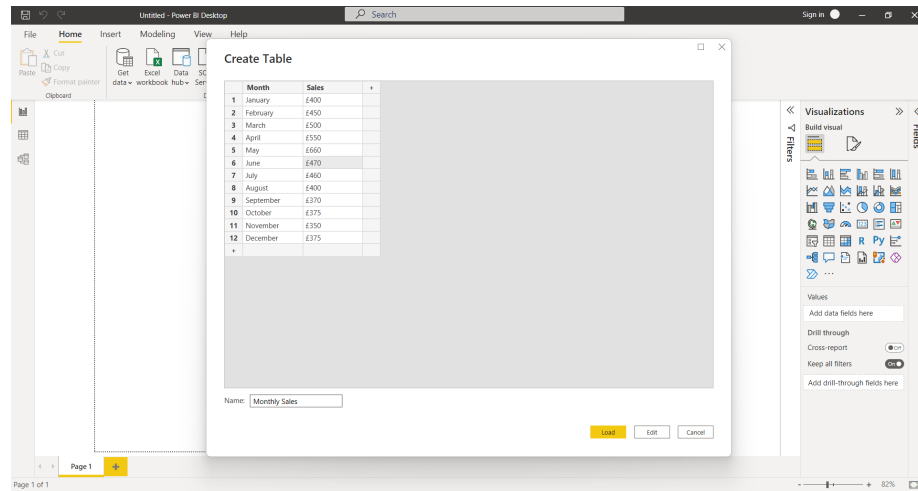
Figure 7.1: Entering data

"Index" and write the number 1 in the first empty row. Write the number 2 in the next row and the number 3 in the next. Continue until each Month has a corresponding numerical index.

When finished, click **OK** then **Close & Apply**.

Go to the data view, click on the "Month" column and find the **Sort by column** option in the **column tools** ribbon and select "Index".

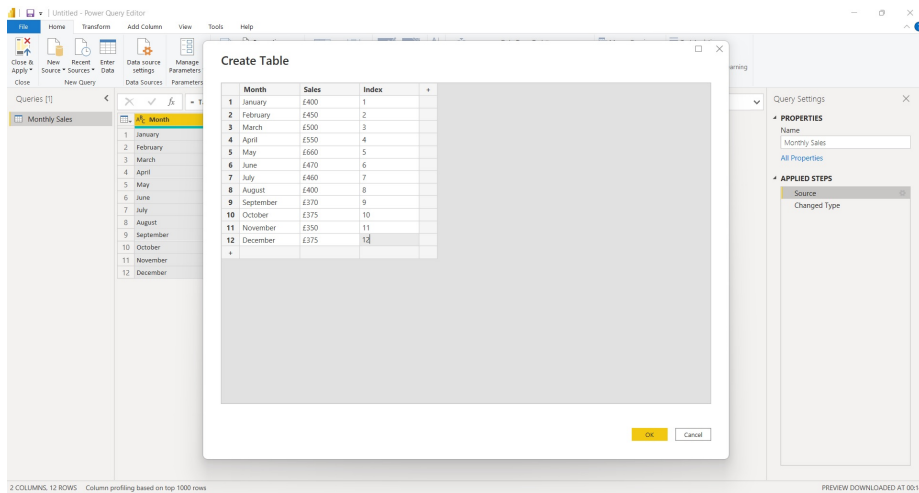Return to the visualisation and the months should now be sorted in the correct order.
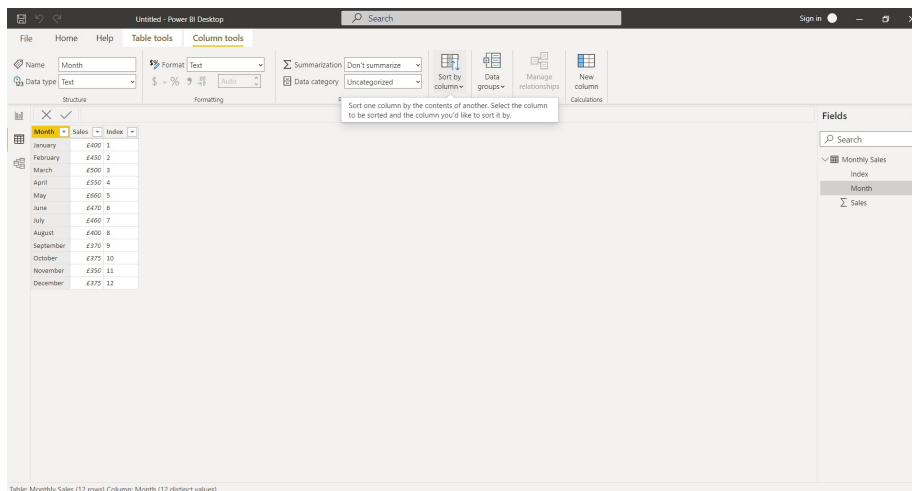
Figure 7.2: Creating an index

Figure 7.3: Sorting by column

Chapter 8

Relationships

Chapter 9

Publishing

———————————————

Chapter 10

Sharing

Chapter 11

Additional Resources

11.1   Additional Resources

This content has been designed to provide a tour of the main features of Power
BI reports. It has skipped a lot of detail to try and give a broad overview
of what is possible with Power BI and provide the basic skills needed to start
experimenting with the software package.

11.1.1   Power BI Documentation

The Power BI Documentation should provide a more in depth look at the funda-
mentals. Microsoft Learn has a number of modules dedicated to Power BI which
are mostly related to Data Analysis learning pathways. Completing these mod-
ules will be useful for consolidating the fundamentals and learning more about
Power BI in greater detail.

11.1.2   DAX

If you're really interested then it might be worth reading into Data Analysis
Expressions (DAX) which is a library of functions that can be used to build
formulas and expressions in Power BI, Analysis Services, and Power Pivot in
Excel data models. This is especially useful in Power Query Editor.