End to End Analytics – Power BI Foundations

Workshop Materials

Minh Pham and Annie Leung
16 November 2023

Contents

Introduction 2
Pre-requisites 2
Exercise 1: My First Power BI 3
Learning Outcomes 3
Part 1: Import CSV File into Power BI 3
Part 2: Setting up Power BI Canvas and Theme 7
Part 2: Add our first Card Visual 10
Part 2: Add our first Clustered Column Chart Visual 14
Part 2: Adding Slicers 20
Exercise 2: Workforce Demographic 25
Learning Outcomes 25
Part 1: Prepare ANZSCO Position Data 25
Part 2: Add ANZSCO Data into Dataset 25
Part 3: Add Generation and Salary Grouping using Power Query 27
Part 4: Creating Explicit Measures 29
Part 4: Creating Visualisation using Measures 33
Exercise 3: Workforce Summary 43
Learning Outcomes 43
Reference Data Model 43
Part 1: Use Power Query to transform the flat file into the data model 43
Part 3: Create Relationships between the tables 47
Part 4: Create and link Date Dimension 50
Part 5: Create Active Employee Snapshot 51
Part 6: Create Measures 55
Part 6: Creating More Visualisations 60
Bonus Exercise: Workforce Trend 71

Introduction

This workshop aims to provide you with the fundamentals of being able to take your own data and be able to use Power BI to create meaningful insights that will be beneficial for your organisation.

We will be going through three main exercises:

- Exercise 1: My First Power BI Report
  In this introductory exercise, we will be looking at the basics of building a simple Power BI report.

- Exercise 2: Workforce Demographic
  In this next exercise, we will be looking at how to enhance our current dataset to be able to get more insights out of our data.

- Exercise 3: Workforce Summary
  On our last exercise, we will be looking at some more advanced techniques in modelling and transforming the data to be able to create more robust visualisations.

For all these exercises, we will be looking at the importance of structuring our datasets right to make visualisations easier to create.

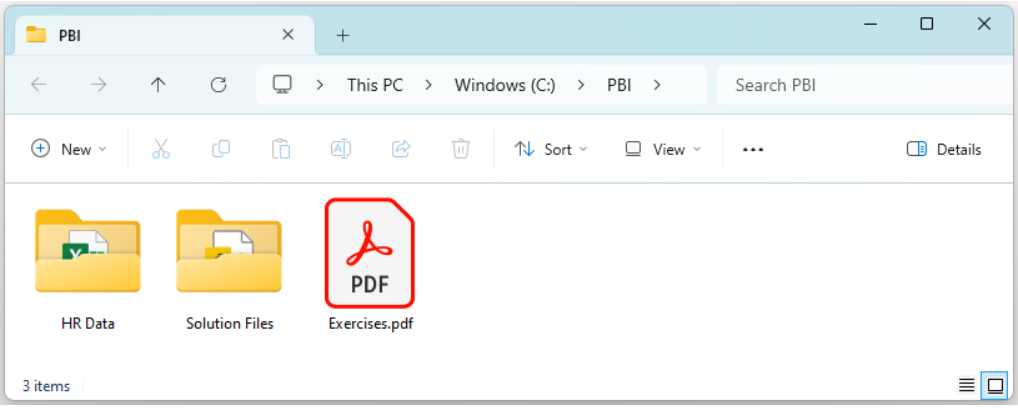
Pre-requisites

Before starting with any of the exercises, you will need to install Power BI, and download the sample datasets in which you will be building your Power BI reports from.

Power BI Desktop: https://www.microsoft.com/en-us/download/details.aspx?id=58494

Exercise Files: You will be provided with the exercise files needed for this workshop. Extract "PBI.zip" to the following file path:

C:\PBI\



Exercise 1: My First Power BI

Learning Outcomes

- Components of a Power BI Report – Dataset vs Visualisations.
- How to import data into Power BI for visualisation
- Basic cleaning and transformation of data
- Type of visualisations and insights that can be created with minimal data manipulation

Part 1: Import CSV File into Power BI

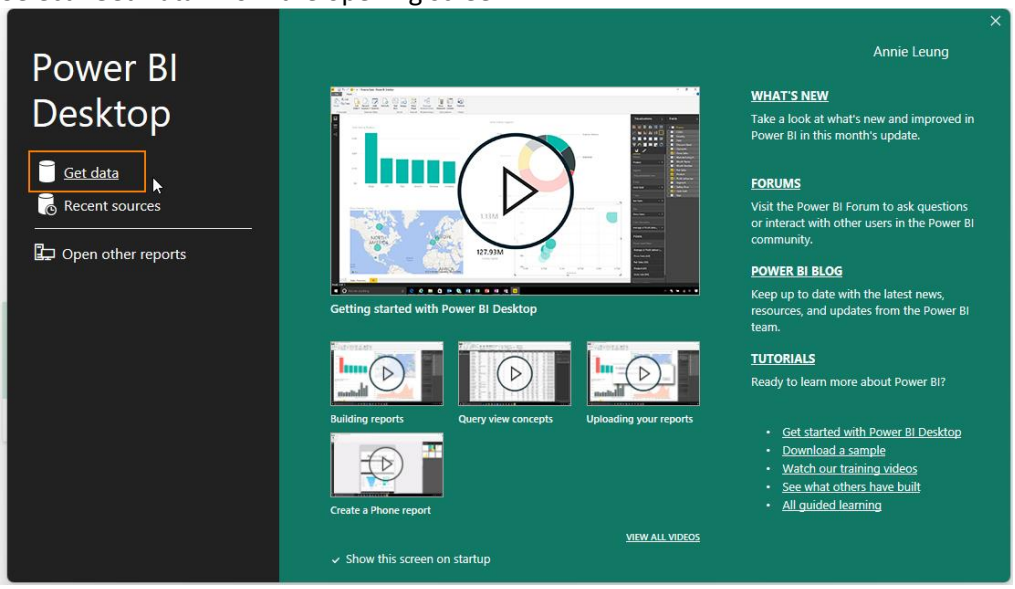
With our HRDataset-Full.csv saved onto our computer, we want to import this data into Power BI to be able to create some visualisations.

In this section, we'll learn to

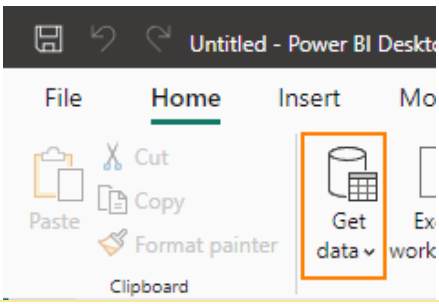
- **Import a CSV file into Power BI**

To do so:

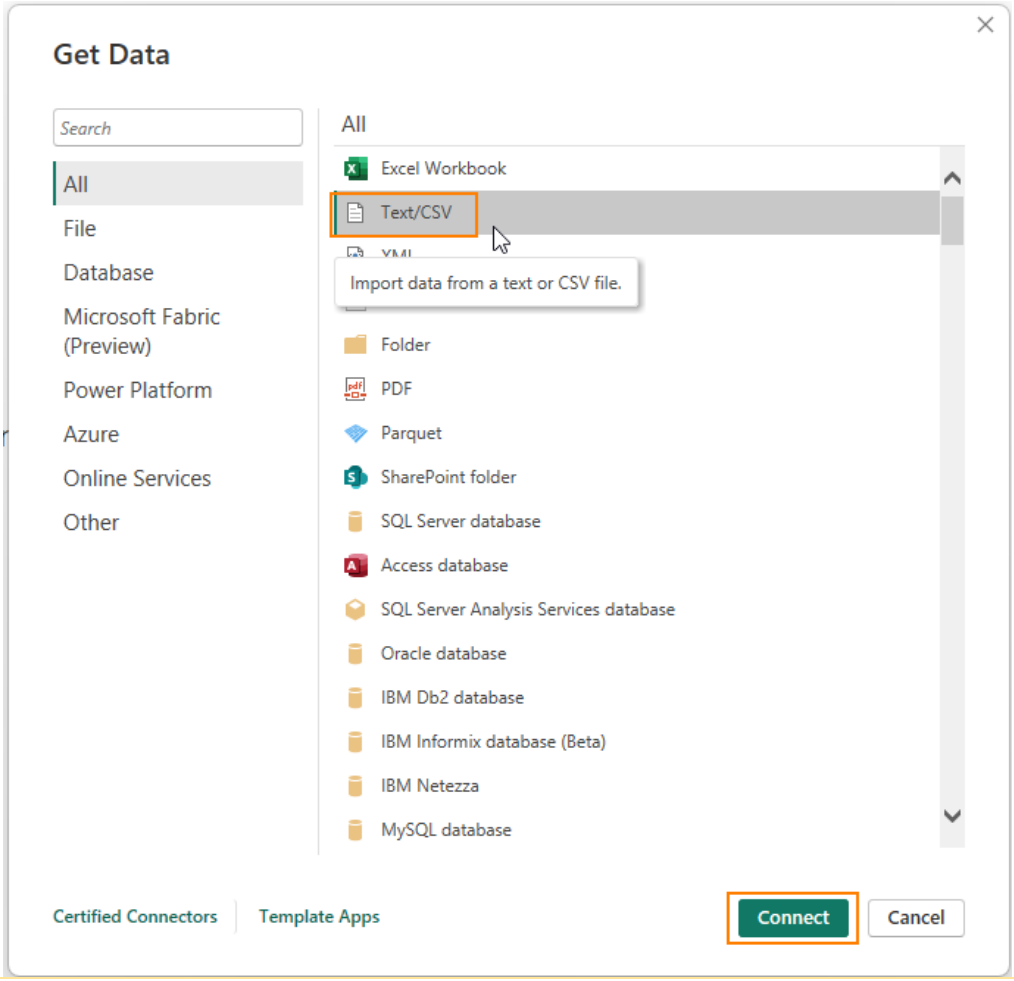
1. Open Power BI Desktop
2. Select "Get Data" from the opening screen:



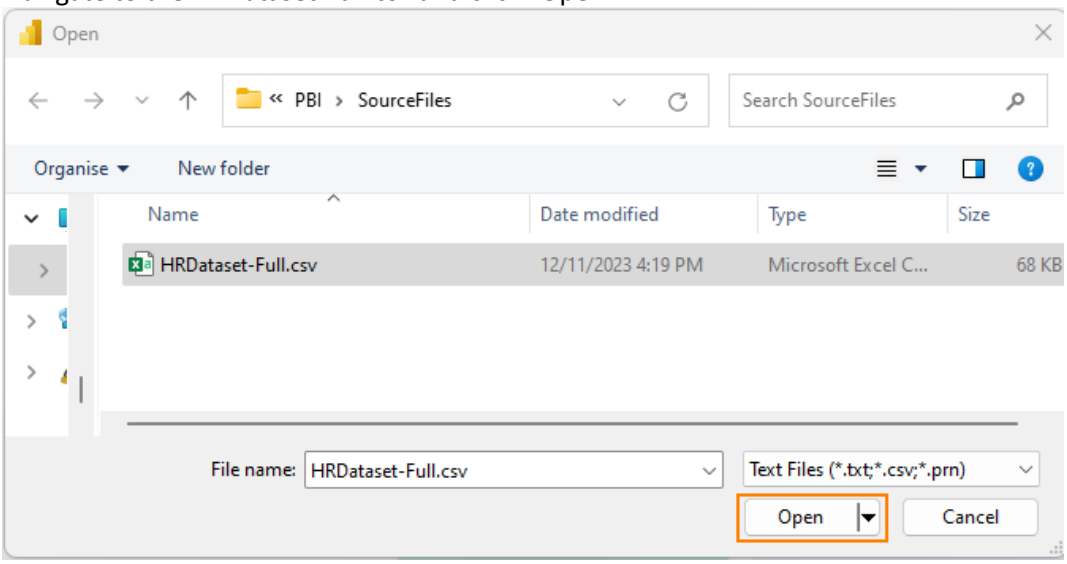
If the opening screen doesn't come up, you can also click on "Get data" from the Home ribbon:



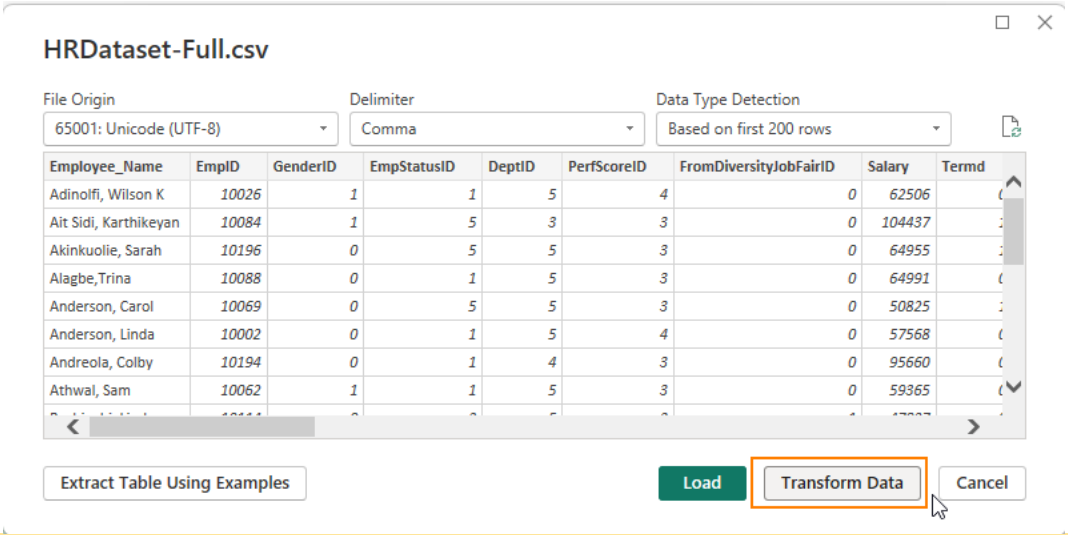
3. Select “Text/CSV”, then press “Connect”



4. Navigate to the HRDataset-Full.csv and click “Open”



5. Take a look at the data in the sample file, then click on “Transform Data”

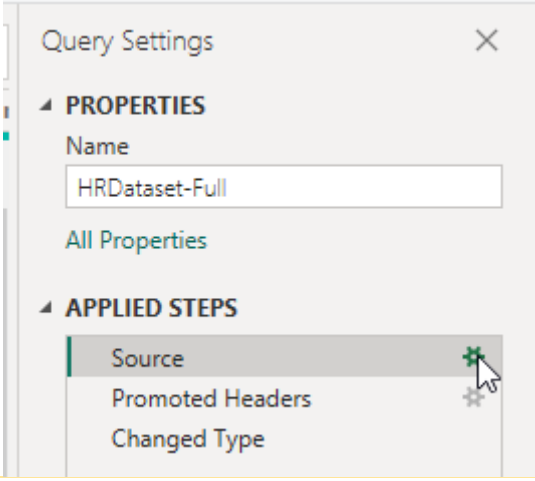


Employee_Name	EmpID	GenderID	EmpStatusID	DeptID	PerfScoreID	FromDiversityJobFairID	Salary	Termd
Adinolfi, Wilson K	10026	1	1	5	4	0	62506	
Ait Sidi, Karthikeyan	10084	1	5	3	3	0	104437	
Akinkuolie, Sarah	10196	0	5	5	3	0	64955	
Alagbe,Trina	10088	0	1	5	3	0	64991	
Anderson, Carol	10069	0	5	5	3	0	50825	
Anderson, Linda	10002	0	1	5	4	0	57568	
Andreola, Colby	10194	0	1	4	3	0	95660	
Athwal, Sam	10062	1	1	5	3	0	59365	

This now takes you to the Power Query Editor, which allows you to do some basic manipulation of the data. We will go through some more advanced transformations in later exercises.

For now, let’s look at what each Applied Step do

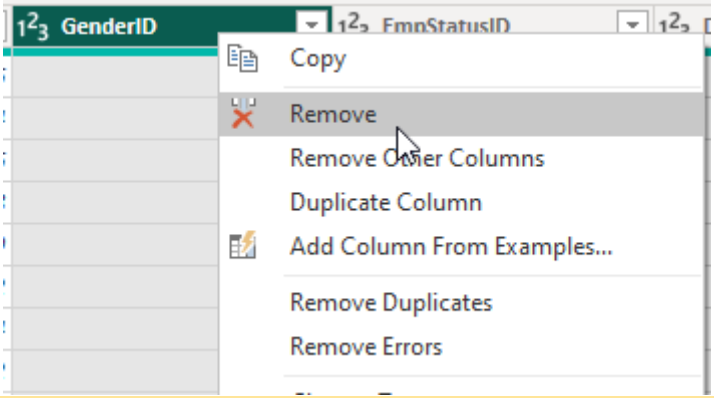
6. In the Query Settings on the right pane, click through each step and take a look at what each of them are doing. Click on the cog icon to see the configuration in more detail.



A lot of the times with system extract files, they will bring in some columns that is part of the system, and not really meaningful for us.

We may want to remove these columns so they don’t confuse the dataset.

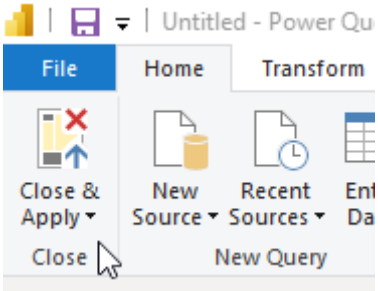
7. Click on the last step “Change Type” to go to the last step, then right click on “GenderID” and click “Remove” to remove this column.



8. Repeat the previous step for the following columns:
   - EmpStatusID
   - DeptID
   - PerfScoreID
   - FromDiversityJobFairID
   - Termd
   - PositionID
   - ManagerID

   Take a look at the last step of the applied step and see how the formula has updated for each one you remove.

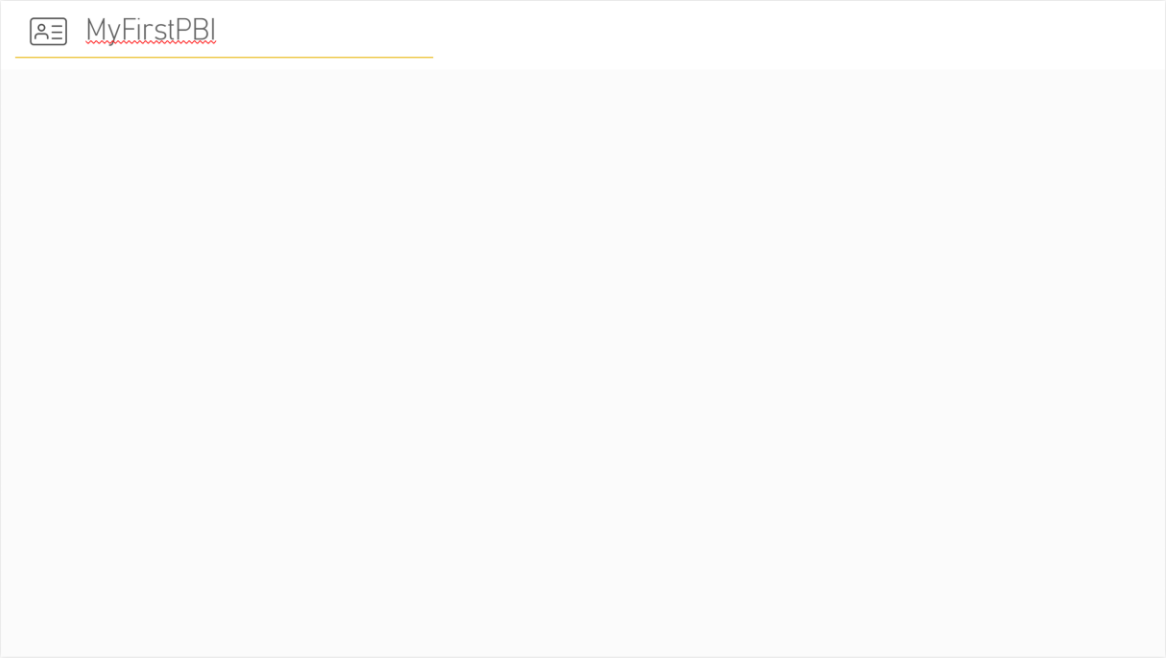
9. Click “Close & Apply” to go back to the Power BI Desktop view. This will now load the data into your Power BI file



10. Press the “Save” button on the top left and save this file as “HR Report.pbix”

Part 2: Setting up Power BI Canvas and Theme

We want to change our Power BI Canvas and colour scheme of our PBIX file so that it is consistent with the company's branding guidelines.



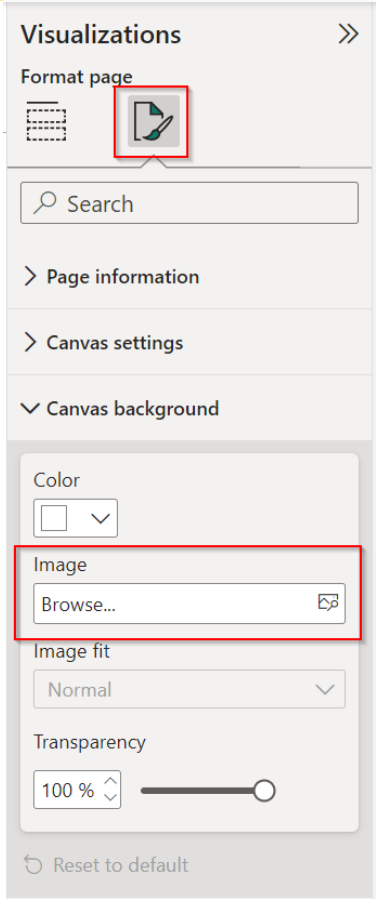
In this section, we'll learn to

- **Import Background to the Power BI Canvas**
- **Import the Colour Scheme using a JSON Theme file.**
- **Add Textbox for our Header Title**
- **Add image to our Canvas**

It is common for companies to develop a PowerBI template that will have these items already loaded. These templates will often have example style guides for the visualisations and headers. Providing these templates helps maintain consistency in design and helps business users meet branding guidelines.

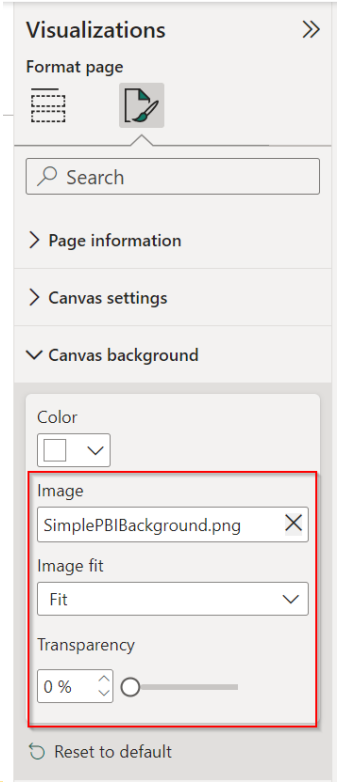
To do so:

1.	Navigate to the **Format Page** option and expand out the **Canvas Background** option



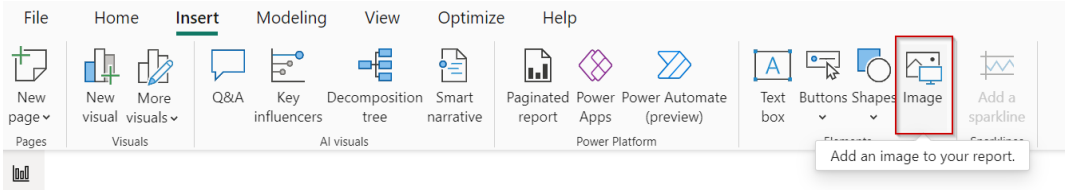
2.

- Browse and select the "**SimplePBIBackground.png**"
- Set Image Fit to "**Fit**"
- Set Transparency to 0%



3. Next, import the elements we need to for the Header Icon

Navigate to **Elements** -> **Image** from the top menu.

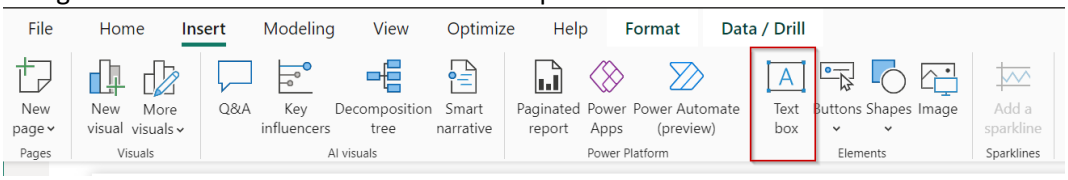


4. Import the **id-card.png** file from the Exercise 1 folder.

Resize the image and position it to the top left-hand corner of the header.

5. Next, we will add the text box for the Header Title of our report.

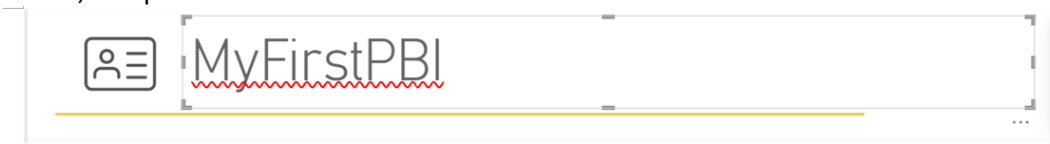
Navigate to **Elements** -> **Text Box** from the top menu.

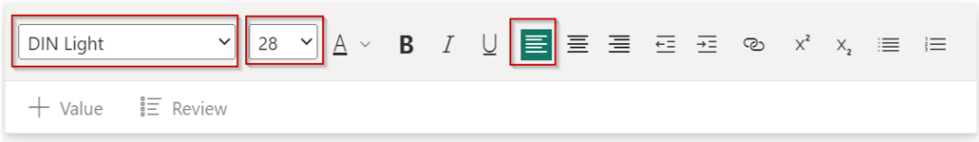


Enter “MyFirstPBI” into the Textbox, and update the Style to match the following:

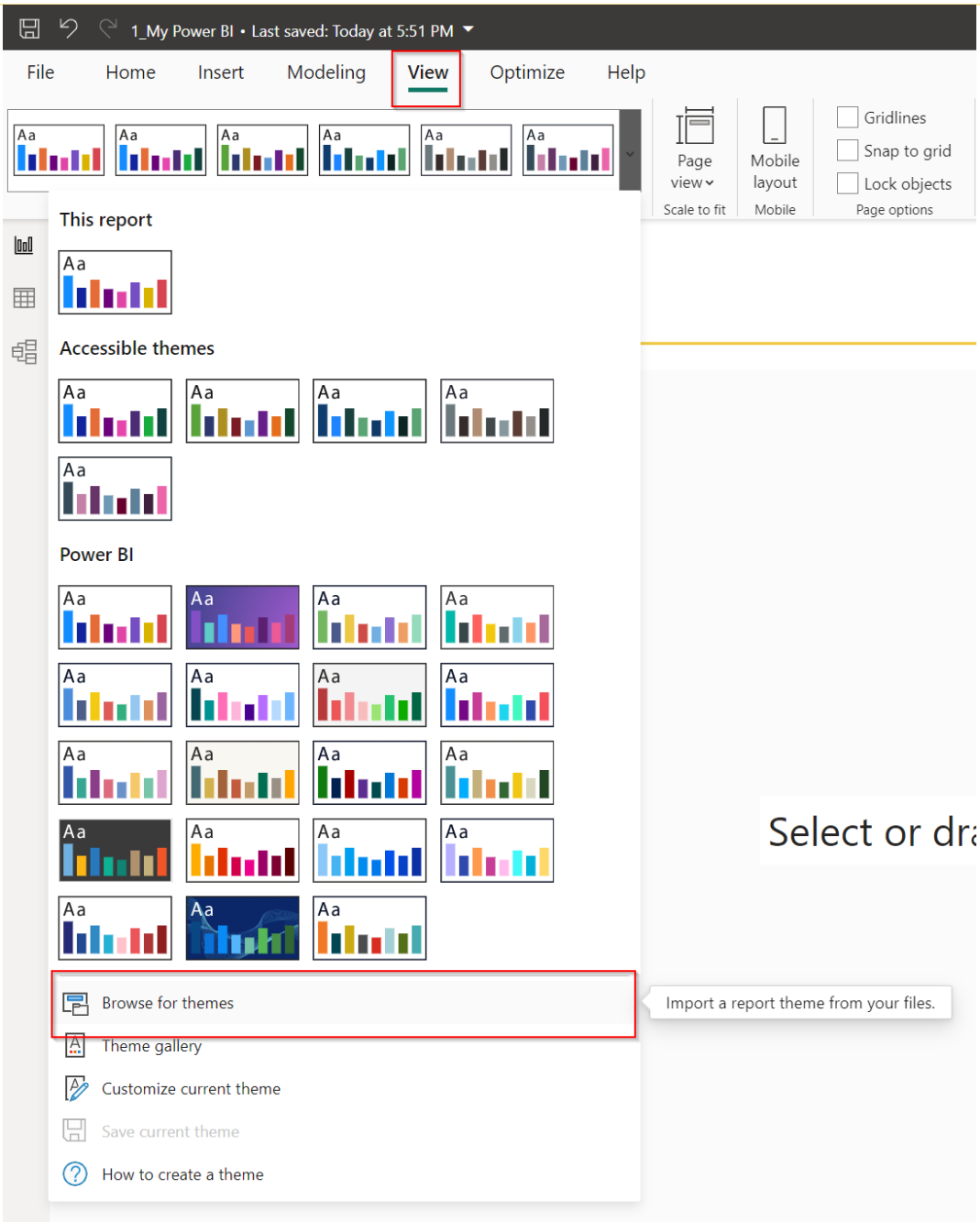
MyFirstPBI
Font: Din Light
Font Size: 28
Alignment: Left Align

Resize, and position he textbox as shown here:





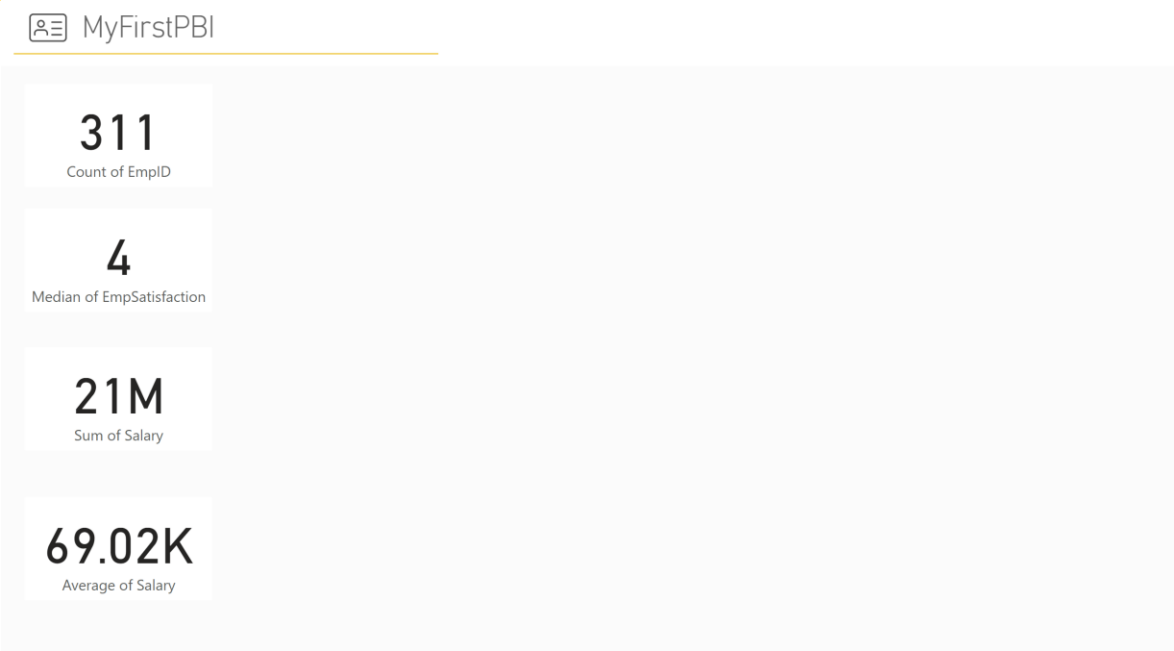
6. To change the Power BI Theme, navigate to the **View** option at the top menu.



7.	Select **Browse for themes** and select the **HRTheme** file from the MyFirstPBI exercise folder. This will change the colour scheme used in the visualisation.

Part 2: Add our first Card Visual

Card Visuals allow us to display a single data point. They're great when you need to quickly convey an important point, and are often used to display KPIs, or a high-level summary of a metric.



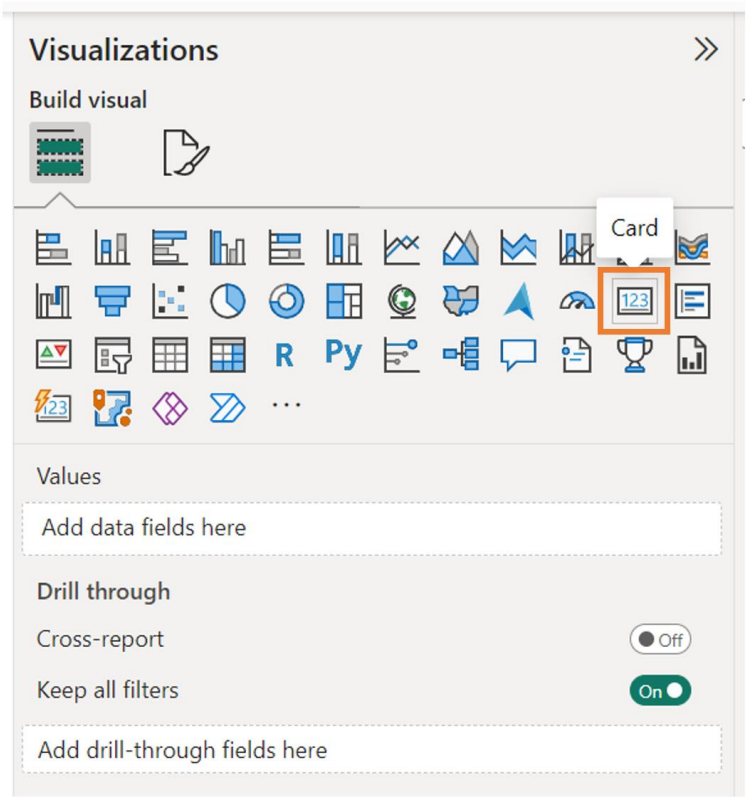
In this section, we'll learn to

- **Create Card Visuals**
- **Use PBI Implicit Measures**

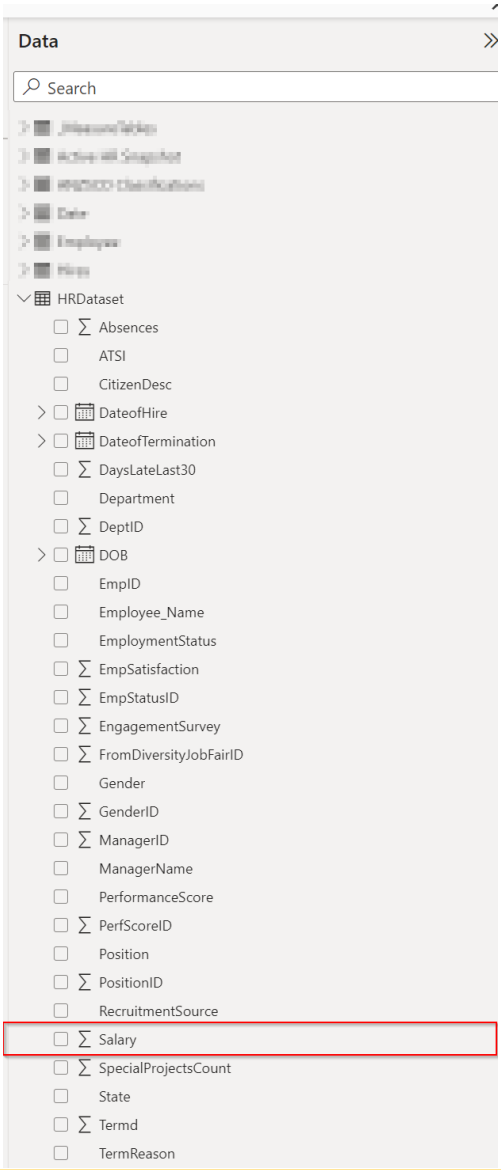
We will start by creating a Card Visual to show us Sum of Salary [Salary]

To do so:

8. Select the **"Card"** Visual from the Visualisation Pane

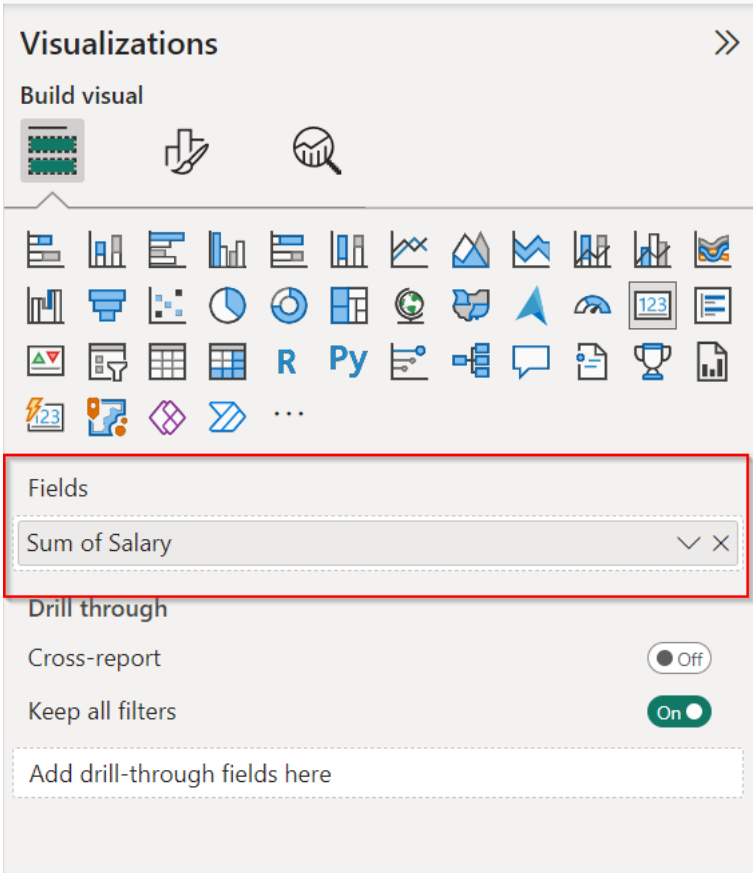


9.	Click on the blank card visual that appears on the Power BI Canvas
10.	Navigate to your **"HRDataset"** table from your **Data** Pane and click on the [Salary] field to add it to your card visual.



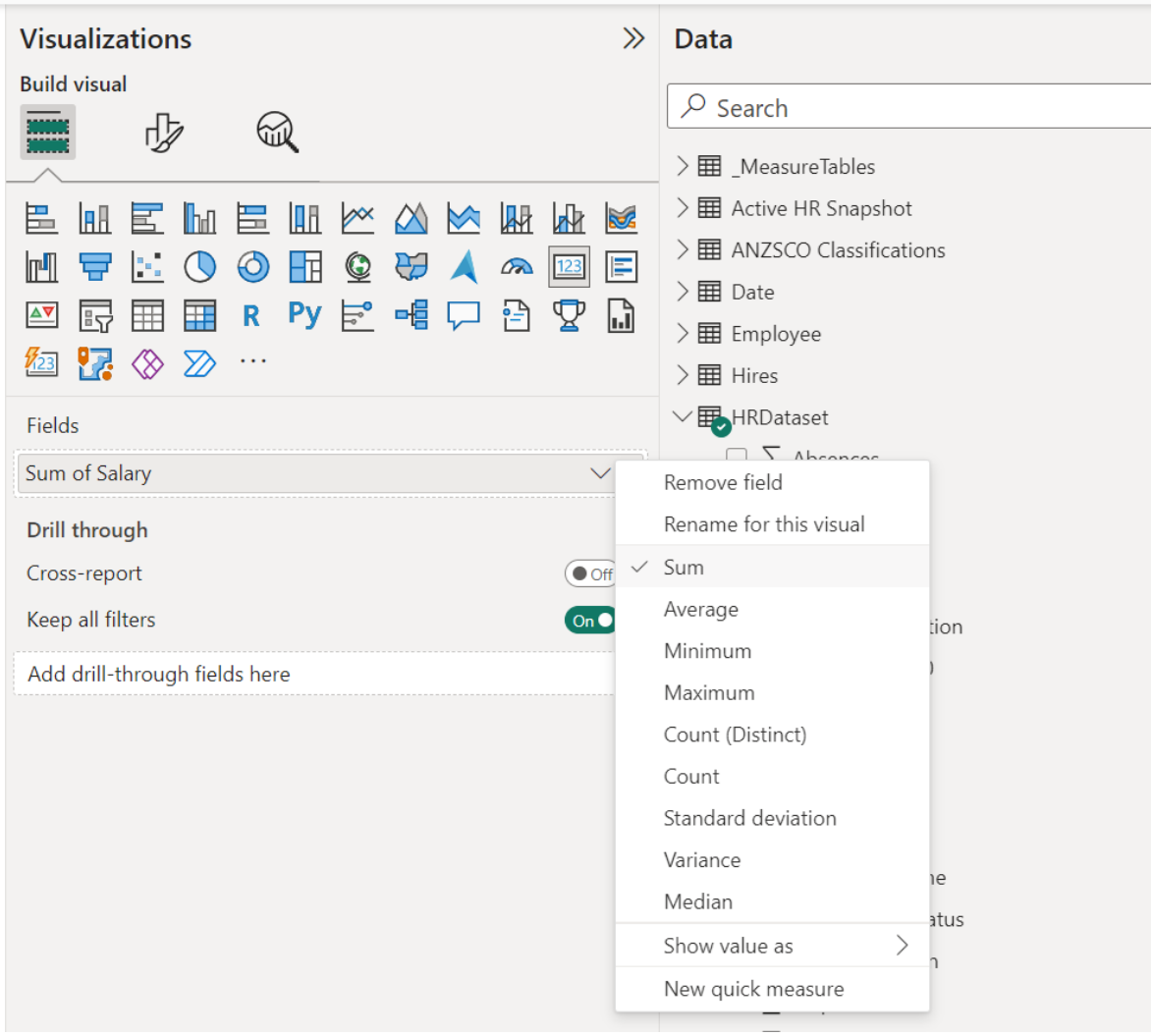
PowerBI will automatically create an "implicit measure" using the [Salary] field.

In the Card Visual shown below, PowerBI has created a "**Sum of Salary**" implicit measure in the **Fields** section.



Implicit measures can be used for performing basic calculations such as counts, sums, and medians of the column selected.

You can change the implicit measure for the field by selecting the "Sum of Salary" measure in the **Fields** section.

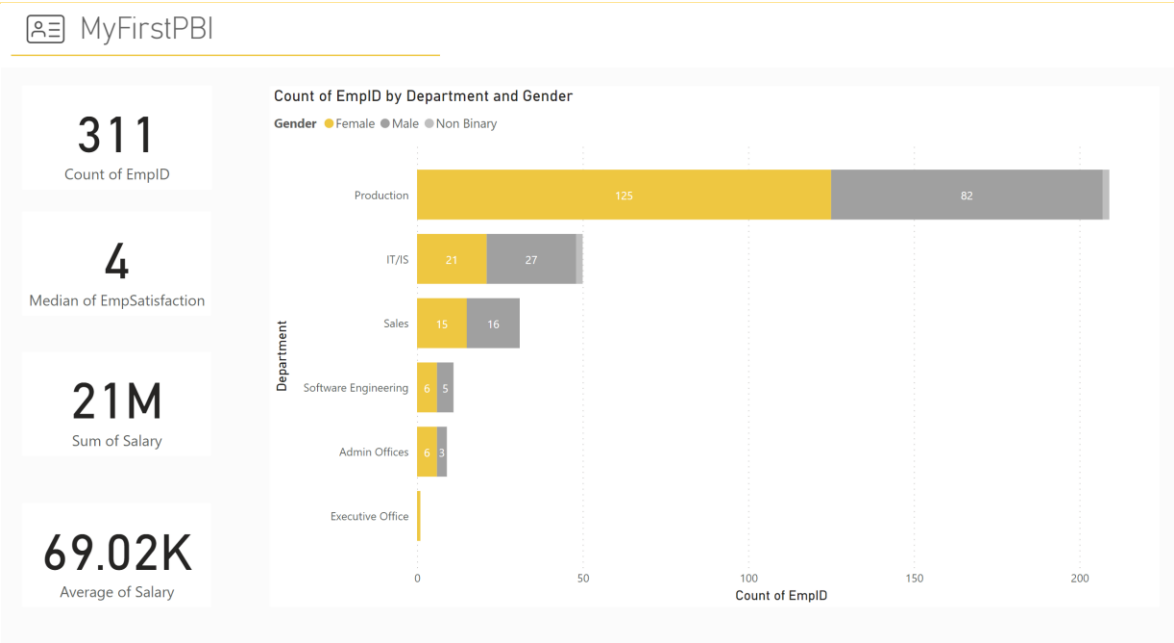


11. Repeat Steps 1 to 3 and create Card Visuals to show the following. The field you need to select from the HRDataset is in [brackets]

- Average Salary [Salary]
- Count of Employee [EmpID]
- Median Employee Satisfaction Score [EmpSatisfaction]

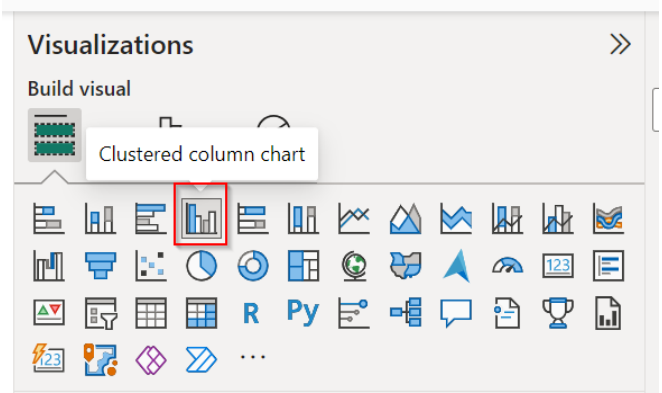
Part 2: Add our first Clustered Column Chart Visual

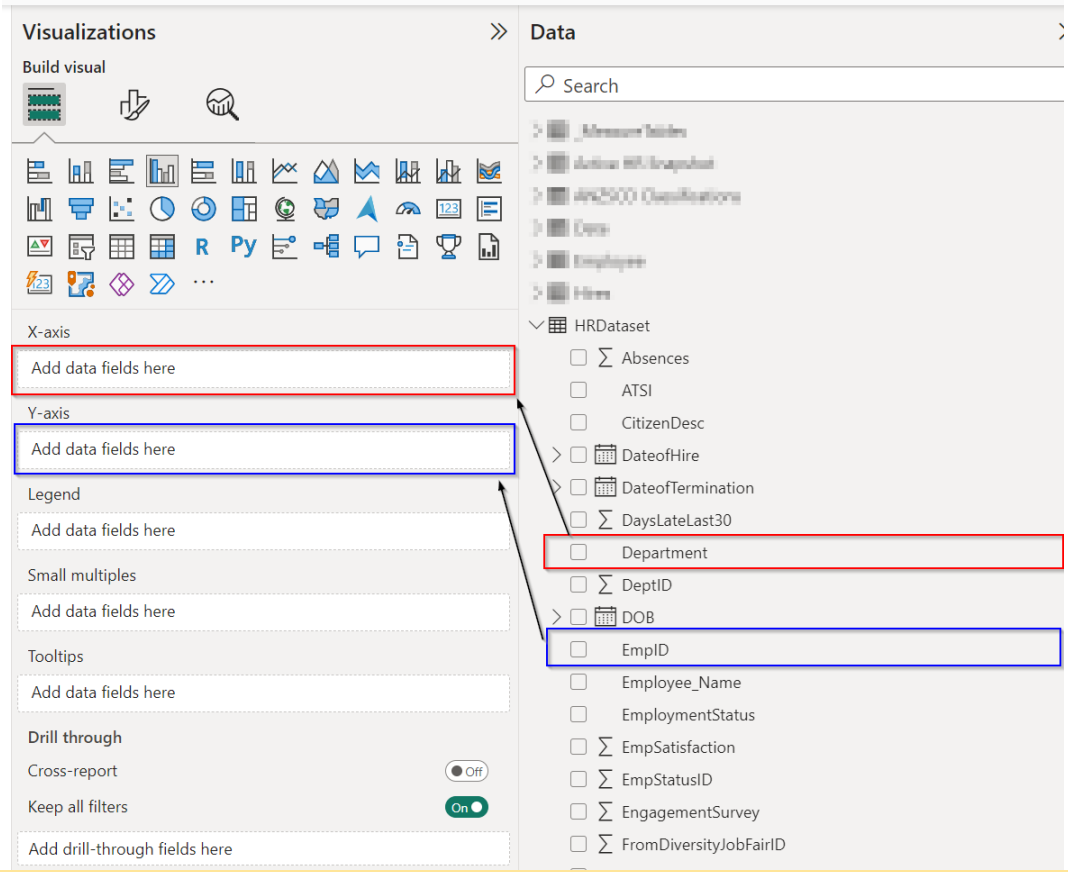
Clustered column charts in Power BI are ideal for comparing multiple categories of data across different groups. In this section, we will be using the chart to show the number of employees that work within each division of the company.



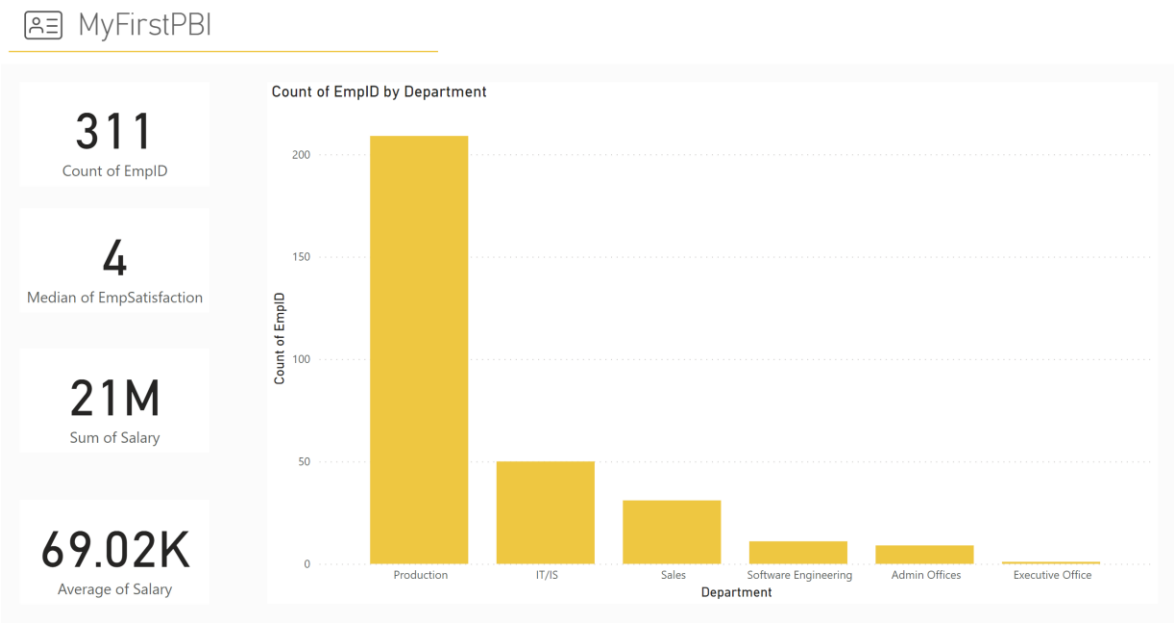
Let's add a "Clustered Column Chart" to our report.

To do so:

1.	Select the **"Clustered Column Chart"** Visual from the Visualisation Pane 
2.	From the **Data** pane, navigate to the **HRDataset** table, and do the following: a) *Drag* the [Department] field to the **X-Axis** b) *Drag* the [EmpId] field to the **Y-Axis**



Check Point: Your PowerBI Canvas should be like the one shown here:

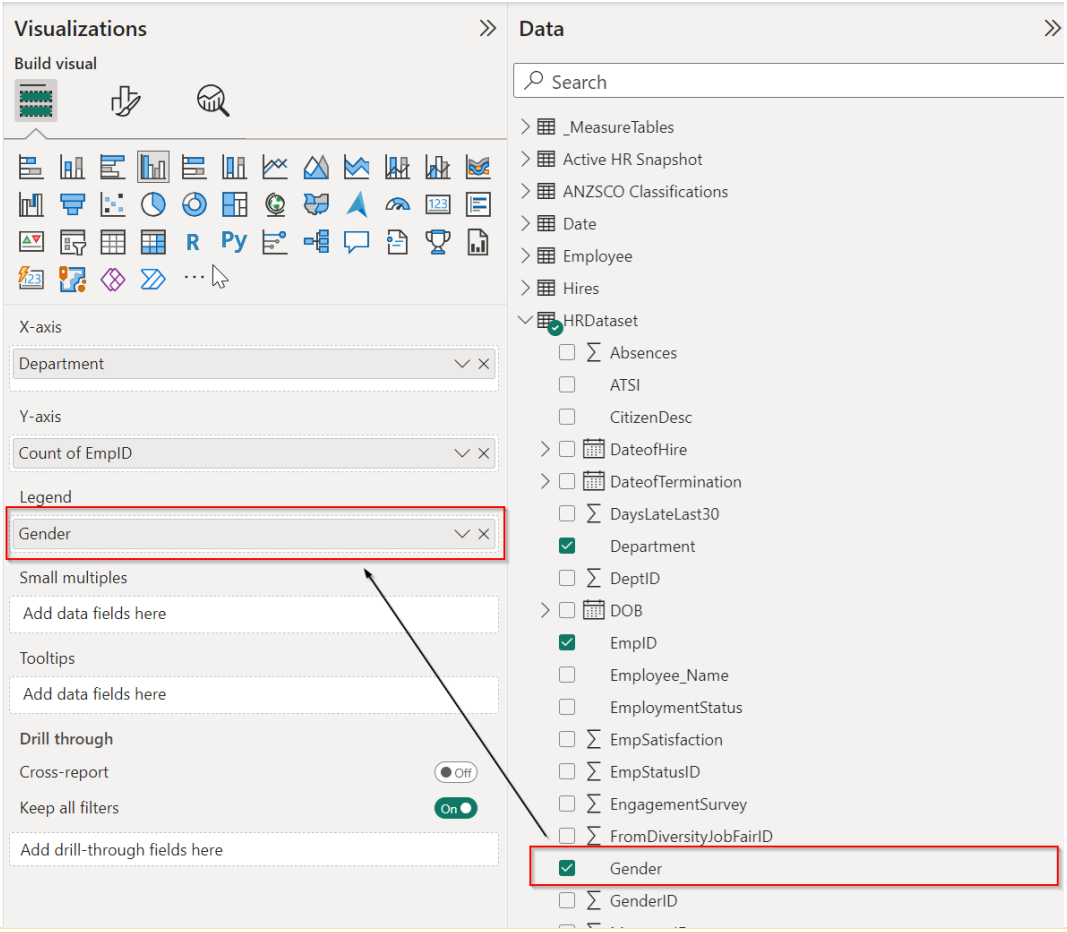


You can add a field to the **legend** of the **Clustered Column Chart** to provide additional detail on the count of employees. Categorical fields such as [Gender], [State] and [ATSI], where the number of distinct values is lower, works best in the **legend** field.

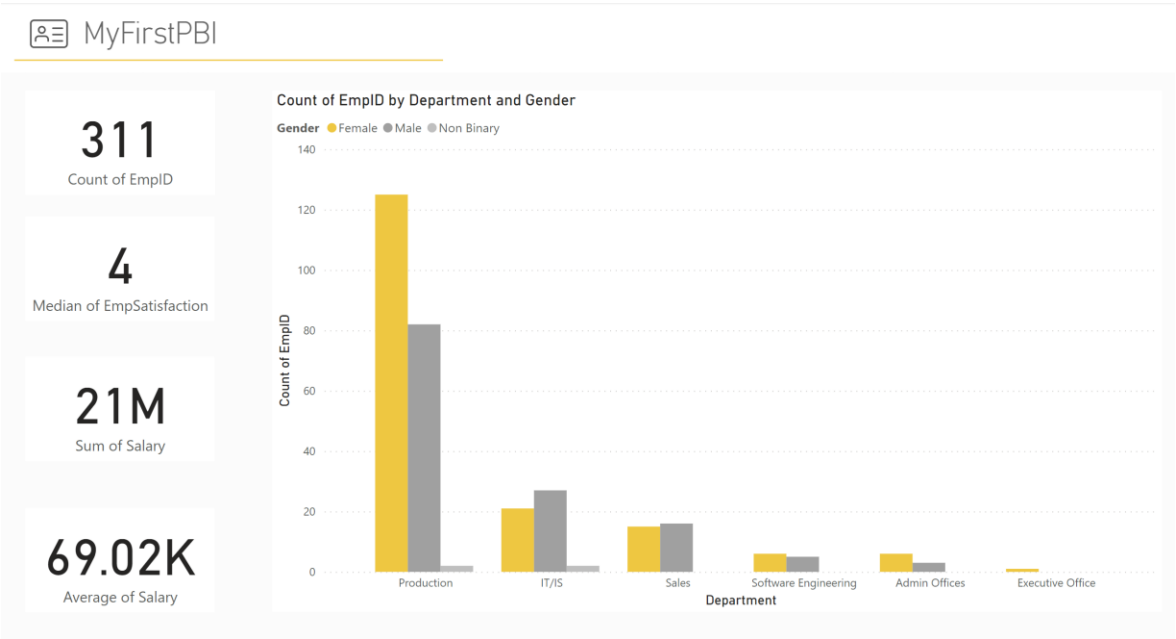
Let's add the [Gender] field to our **Clustered Column Chart**

3.	From the **Data** pane, navigate to the **HRDataset** table, and do the following:

a) *Drag* the [Gender] field to the **X-Axis**

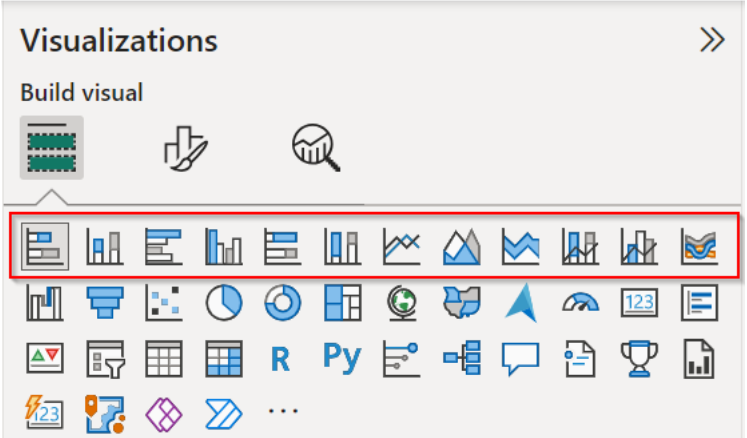


Check Point: Your PowerBI Canvas should be like the one shown here:



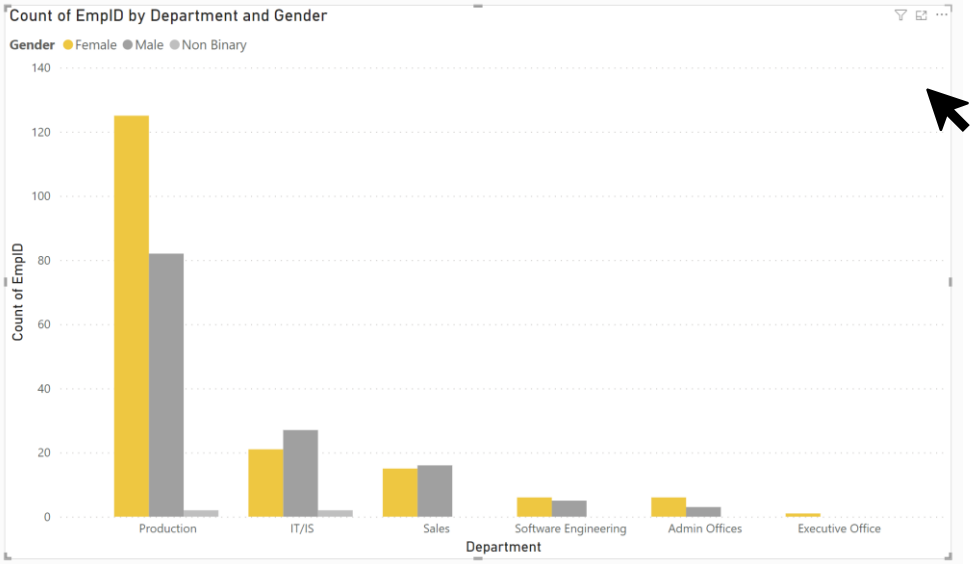
The Bar and Column Charts available in Power BI all use the same parameters (eg. X-Axis, Y-Axis, Legend). This means you can easily switch between these chart designs without having to modify any of the fields.

The following Visuals all use X-Axis, Y-Axis, and Legend field.

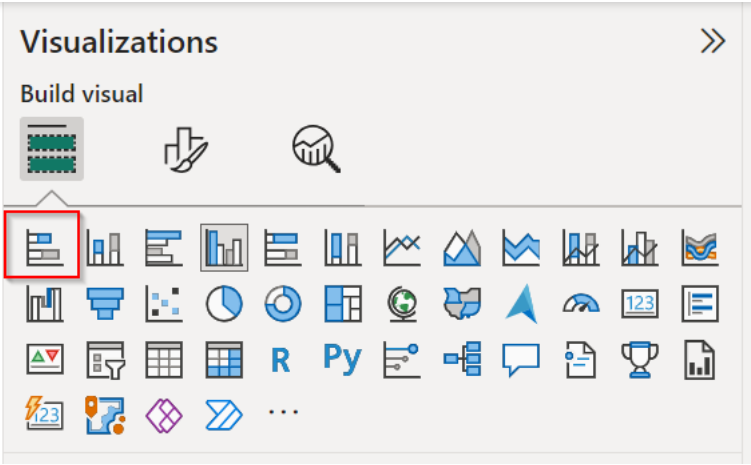


Let's change our chart to a **Stacked Bar Chart**

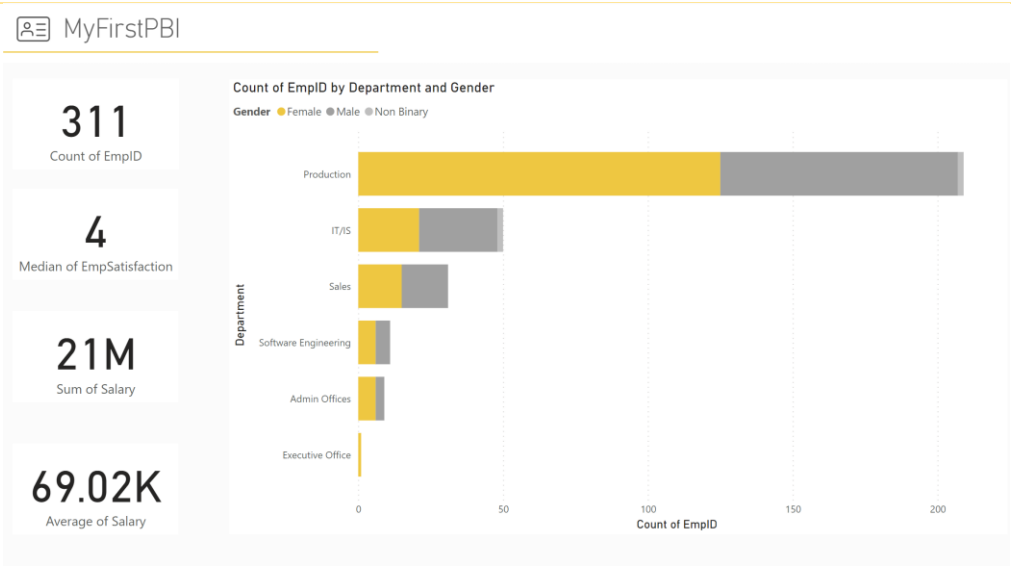
4. Select your existing **Clustered Column Chart** in the Canvas to make it active.



5. From the **Visualisations** Pane, select the **Stacked Bar Chart** visual.

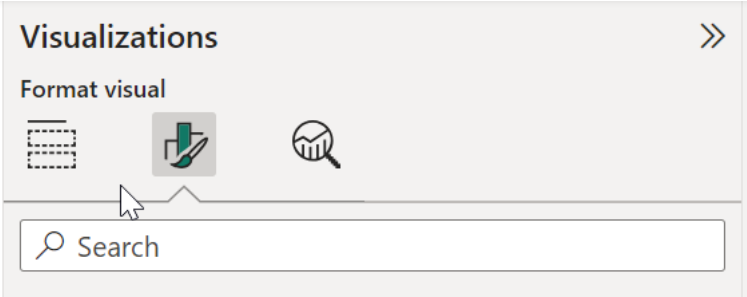


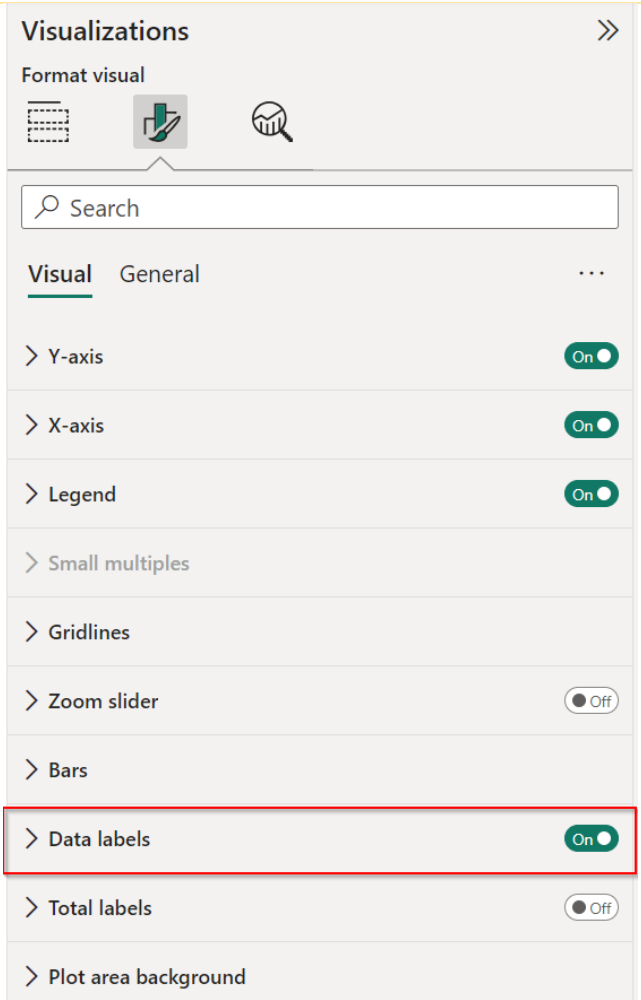
Check Point: Your PowerBI Canvas should be like the one shown here:



We can also add **Data Labels** to our chart.

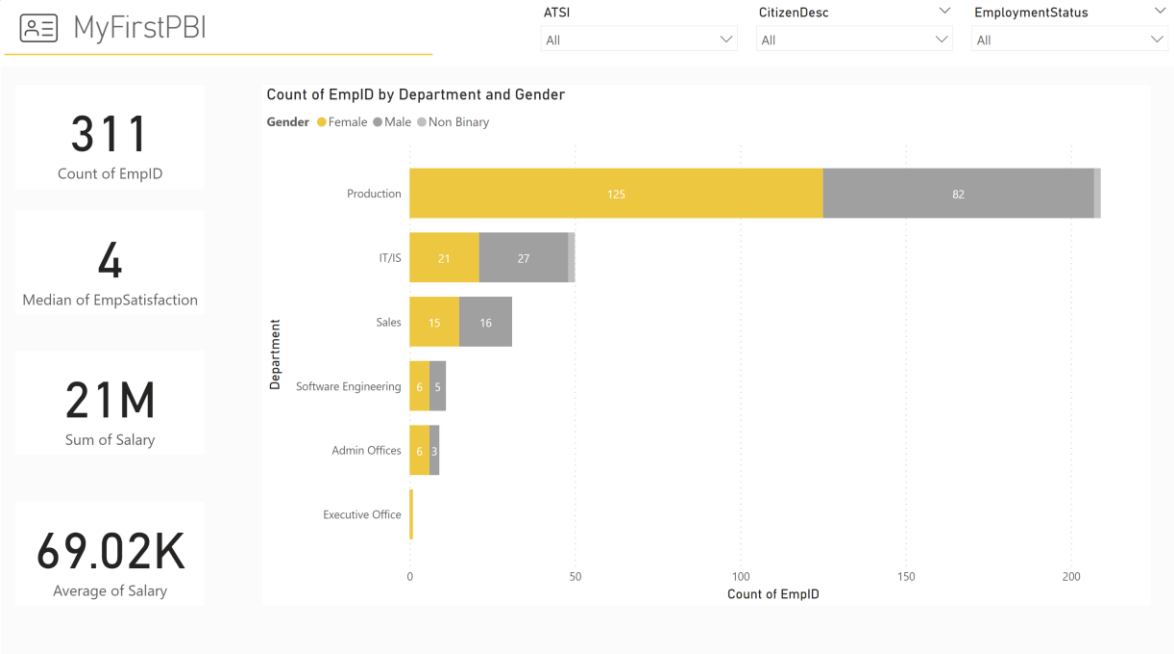
To do so:

6.	With the Chart selected, navigate to the **Visualisation** Pane, and select the **Format Visual** Icon.  This will open up the format options available for that visual.
7.	Navigate to the **Data Label** option, and toggle it **ON**



Part 2: Adding Slicers

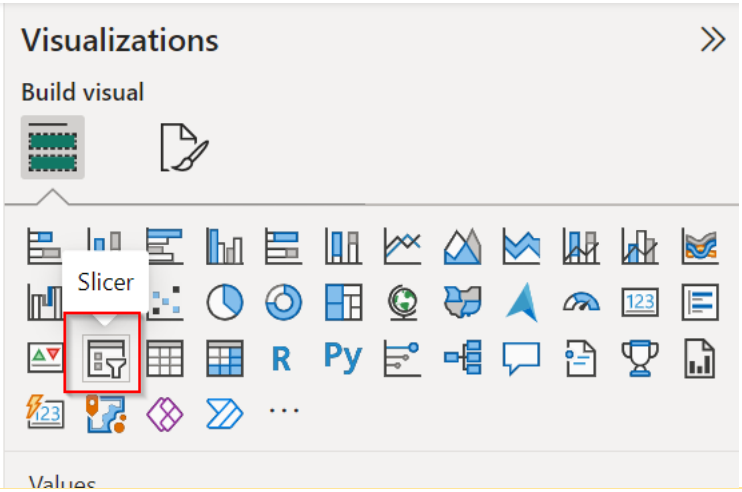
Slicers are a visual filter that allow you to filter the data that is used to create the visualisations in Power BI.

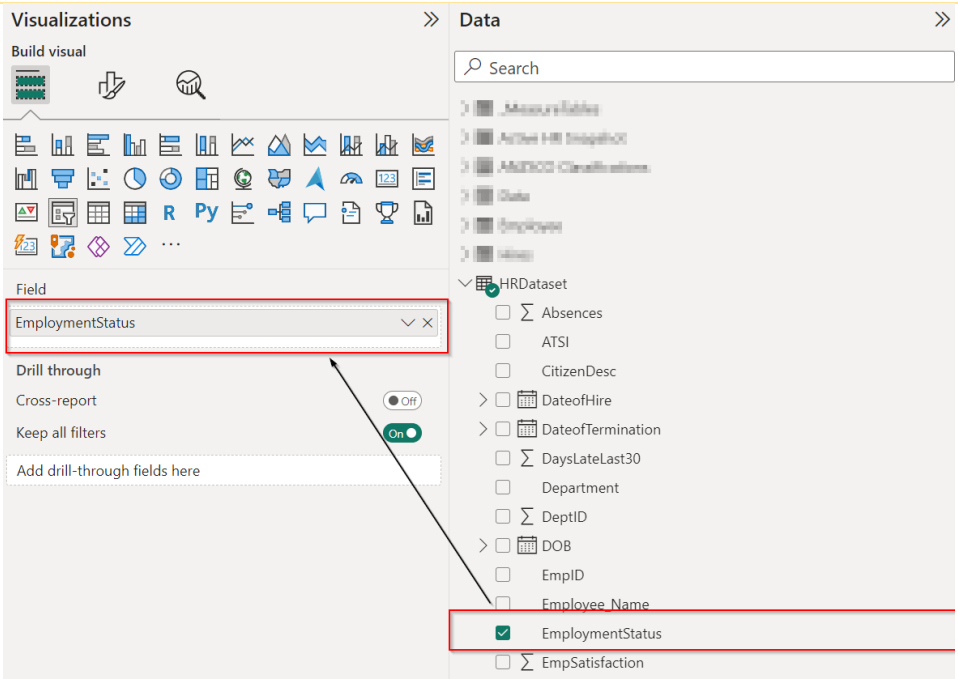


We will start by adding a Slicer with the [EmploymentStatus] field. This will allow us to change our visualisations to focus on employees that have the following [Employment Status]

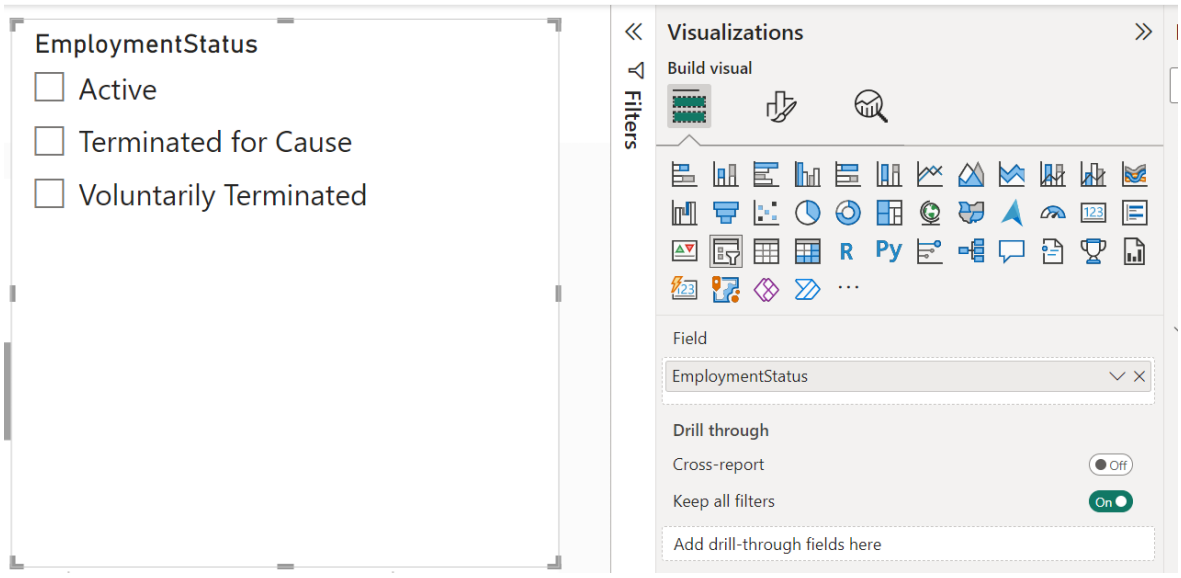
- **Active**
- **Terminated for Cause**
- **Voluntarily Terminated**

To do so:

1.	Select the **"Slicer"** Visual from the Visualisation Pane 
2.	From the **Data** pane, navigate to the **HRDataset** table, and do the following: a) *Drag* the [EmploymentStatus] field to the **Field** area



Check Point: Your **Slicer** should look like the following:

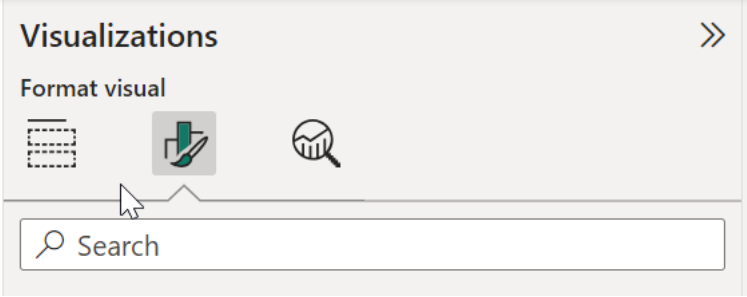


By default, the Slicer Style is set to ***Vertical List*.**

Let's change the style to a drop-down.

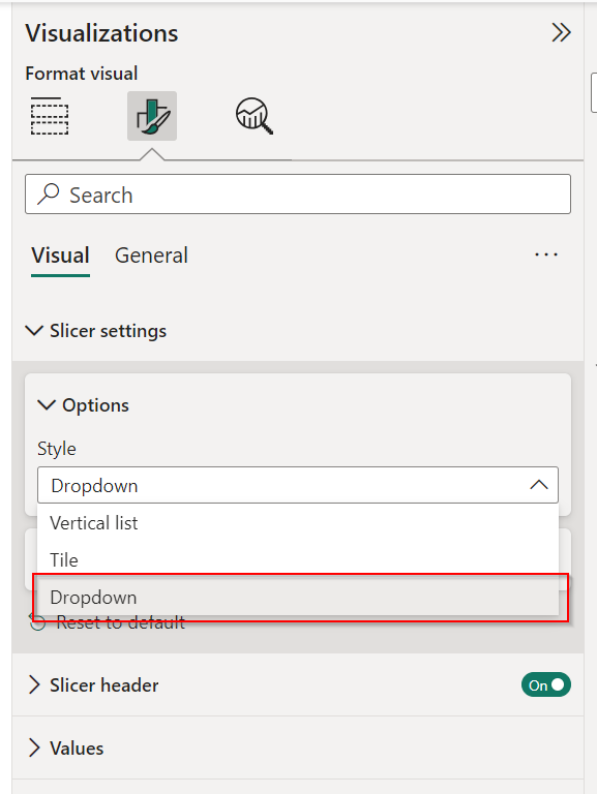
To do so:

3. With the Slicer selected, navigate to the **Visualisation** Pane, and select the **Format Visual** Icon.

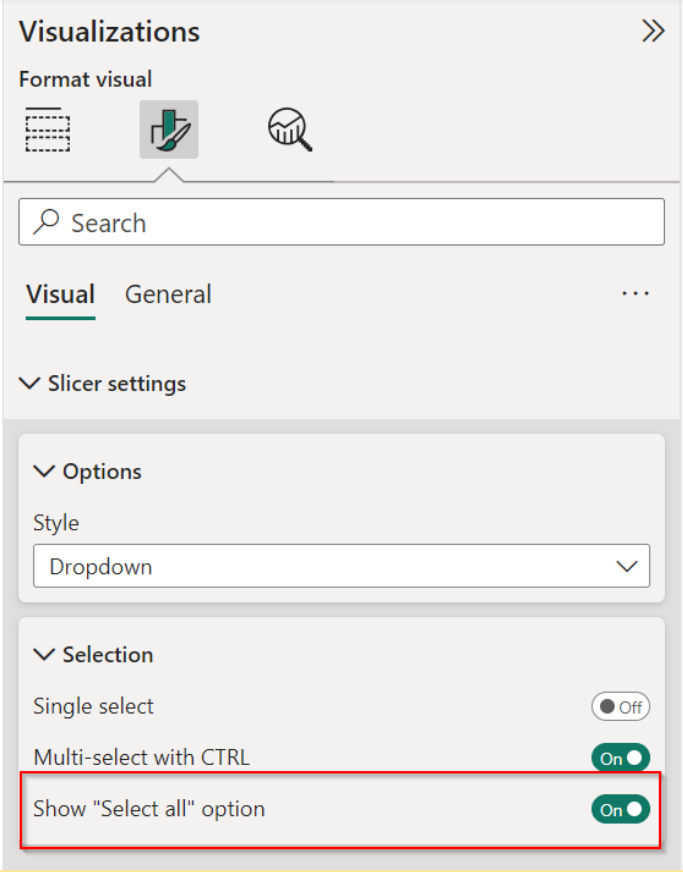


This will open the format options available for that slicer.

4. Expand **Slicer Settings**, and under **Options -> Style** select the **Dropdown** style.

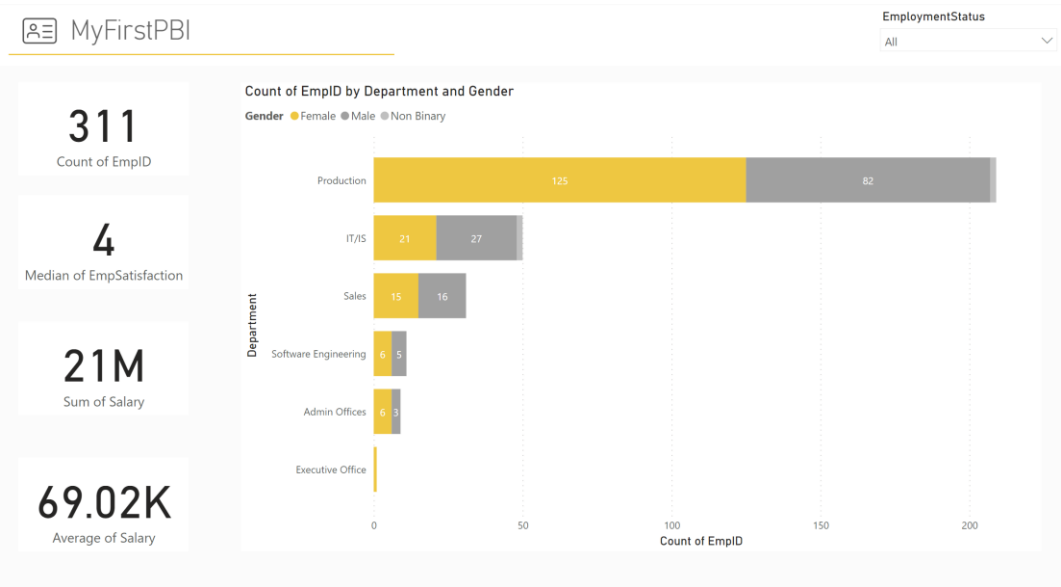


12. We can also enable the "**Select all**" option in our Dropdown by toggling the option under **Slicer settings -> Selection** to ON.



13. Resize the Slicer and position it on the top right side of the report.

Your PowerBI Canvas should be like the one shown here:



You can add multiple Slicers to filter your data with multiple categories.

14. Repeat Steps 1 to 6, and add the following Slicers:
    - [CitizenDesc]
    - [ATSI]

With the Slicers added, your PowerBI report can now answer questions about employees with specific citizenships, employment statuses, and whether they identify as Aboriginal or Torres Strait Islander.

Exercise 2: Workforce Demographic

Learning Outcomes

- Add external sources to enrich your dataset
- Use Power Query to add grouping for generation (age) and salary range
- Create relationships in Power BI
- Create more visualisations using the new data model

Part 1: Prepare ANZSCO Position Data

The ABS releases a list of standard classifications of occupations. This information can be used to enhance your dataset so that it can be benchmarked against other organisations.

ANZSCO - Australian and New Zealand Standard Classification of Occupations, 2022 | Australian Bureau of Statistics (abs.gov.au)

To prepare a list similar to this.

	A	B	C	D
1	Position	Occupation Level 6 (ANZSCO)	Occupation Description (ANZSCO)	Occuption Description Level 1
2	Accountant I	221111	Accountant (General)	Professionals
3	Administrative Assistant	531111	General Clerk	Clerical and Administrative Workers
4	Area Sales Manager	131112	Sales and Marketing Manager	Managers
5	BI Developer	224114	Data Analyst	Professionals
6	BI Director	111211	Corporate General Manager	Managers
7	CIO	135111	Chief Information Officer	Managers
8	Data Analyst	224114	Data Analyst	Professionals
9	Data Architect	261399	Software and Applications Programmers nec	Professionals

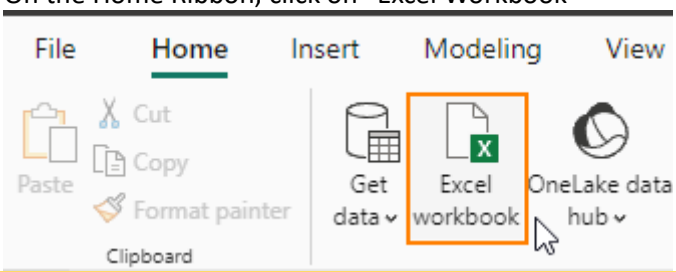
1. Get a unique list of positions from your dataset in Excel
2. Map the positions to their corresponding ANZSCO description and classification

This Excel spreadsheet can now be loaded into Power BI to enhance the dataset

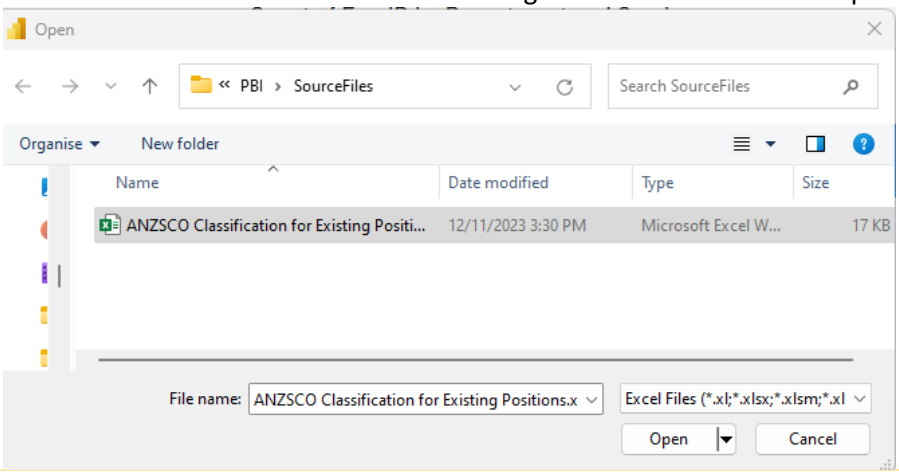
We have prepared a copy of this spreadsheet in the course materials.

Part 2: Add ANZSCO Data into Dataset

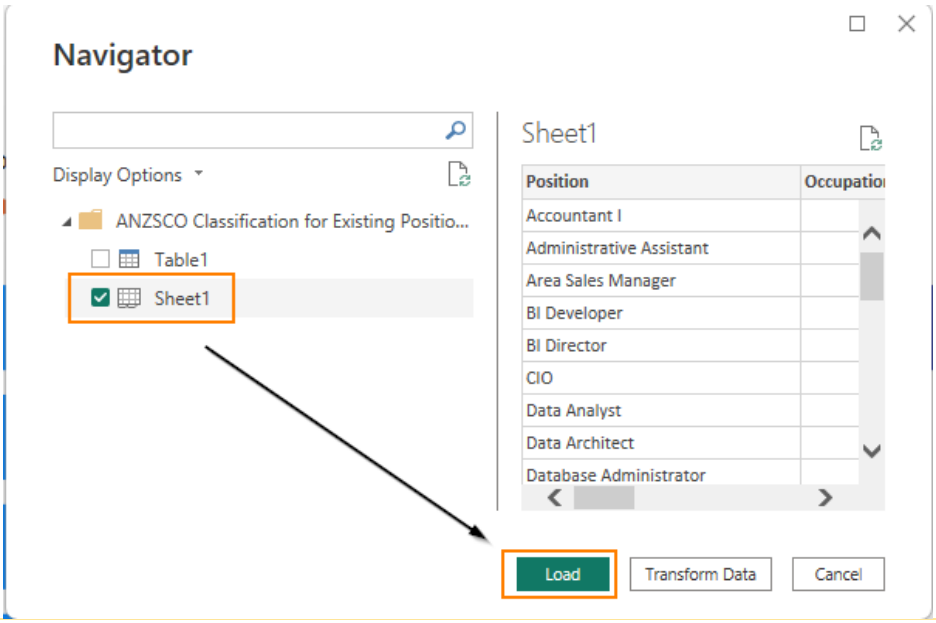
1. On the Home Ribbon, click on "Excel Workbook"



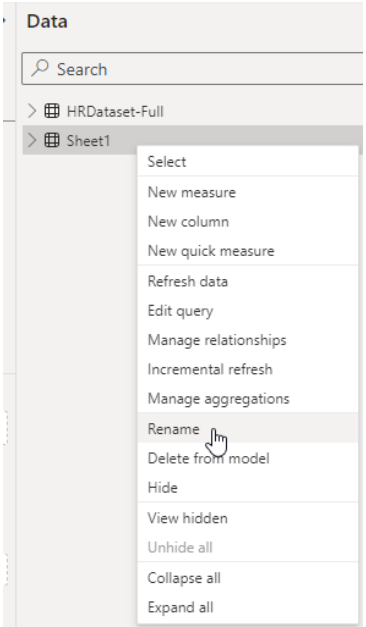
2. Browse to “ANZSCO Classification for Existing Positions.xlsx” and click Open

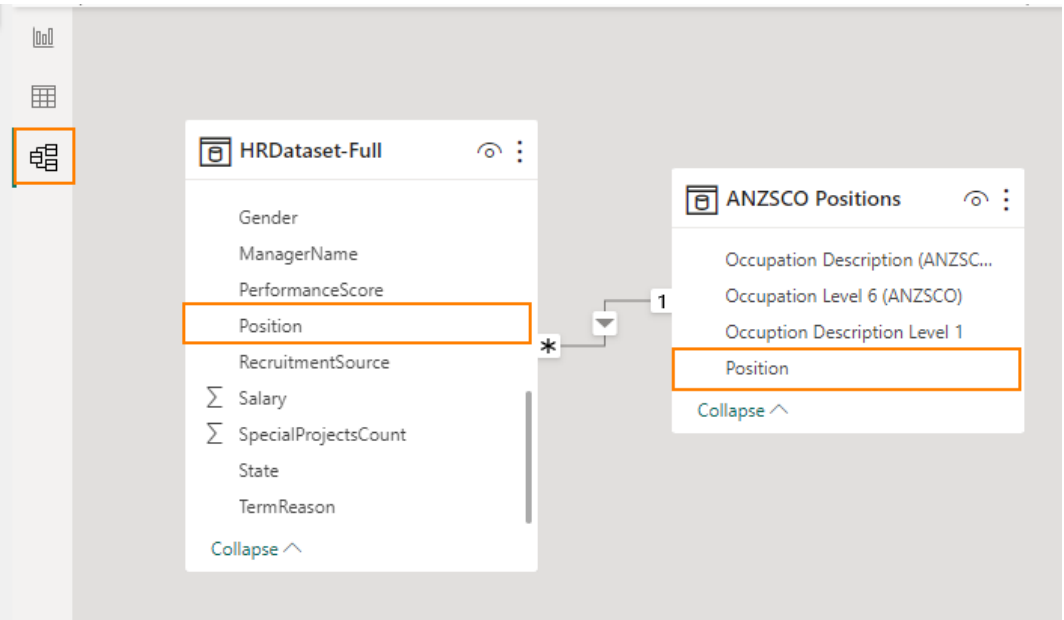


3. Select “Sheet1” and click “Load”

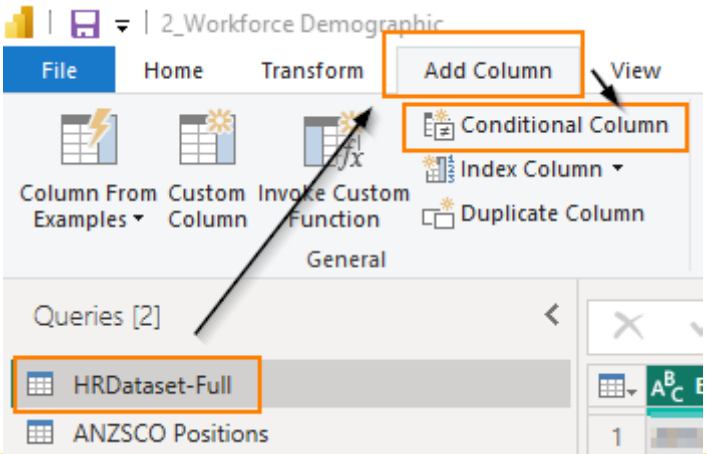


4. On the Data Pane, rename “Sheet1” to “ANZSCO Positions”



5.	Click on the Data Model view icon on the left pane. Check that Power BI has automatically created the Position relationship between the HRDataset-Full and the ANZSCO Positions table. If it hasn't, drag the columns onto each other to create the relationship.  The two tables are now linked together, and you can create visuals that leverage off the extended fields from the ANZSCO positions table.

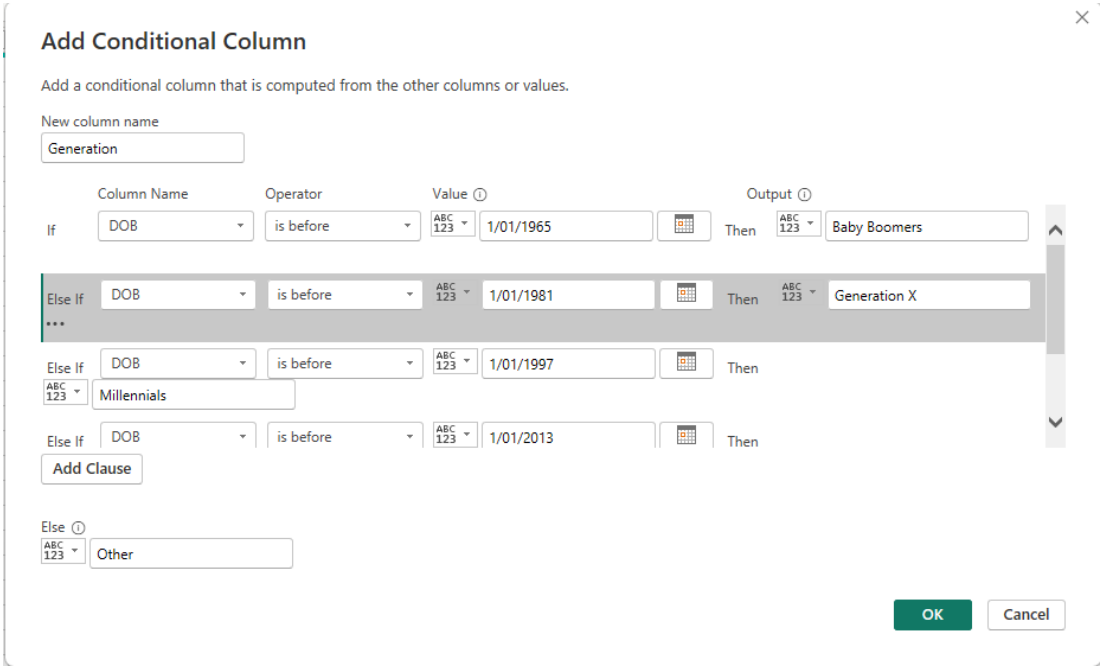
Part 3: Add Generation and Salary Grouping using Power Query

Sometimes, we may want to create custom categories for datasets, such as Age Range, or Salary Range information. We can easily do that with the native capabilities of Power Query.	
Add "Generation"	
1.	Click on "Transform Data" to open the Power Query Editor
2.	Click on "HRDataset-Full" in the Query Pane, then click on "Add Column", "Conditional Column". 
3.	For New Column Name, enter "Generation"

Generation	Start Year	End Year
Baby Boomers	1946	1964
Generation X	1965	1980
Millennials	1981	1996
Generation Z	1997	2012
Generation Alpha	2013	2025

For the condition, add the following:
If DOB is before “1/1/1965” Then “Baby Boomers”
Else If DOB is before “1/1/1981” Then “Generation X”
Else If DOB is before “1/1/1997” Then “DOB”
Else If DOB is before “1/1/2013” Then “Generation Z”
Else If DOB is before “1/1/2026” Then “Generation Alpha”
Else “Other”

Then click OK



4. Try to do the same for Salary Range, with the following categories:

$0 to $50,000
$50,001 to $75,000
$75,001 to $100,000
$100,001 to $125,000
$125,001 to $150,000
$150,001 and over

Part 4: Creating Explicit Measures

Up until this point, we have been using Implicit Measures for our visualisations.

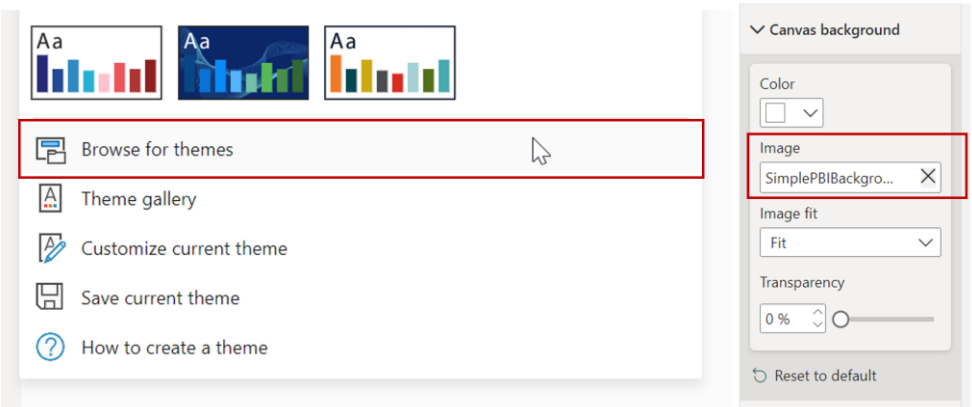
Explicit Measures are measures which are defined by the user through DAX formulas. The main benefit of explicit measures over implicit measures is the greater control and flexibility they offer. Explicit measures allow for more sophisticated and customized calculations.

We will be using DAX functions to perform the following count of employees, and variations of the count based on the employment type of the employee.

1. **Count of Employees**
2. **Count of Employees (Casual)**
3. **Count of Employees (Fixed Term Full-Time)**
4. **Count of Employees (Fixed Term Part-Time)**
5. **Count of Employees (Full-Time)**
6. **Count of Employees (Part-Time)**

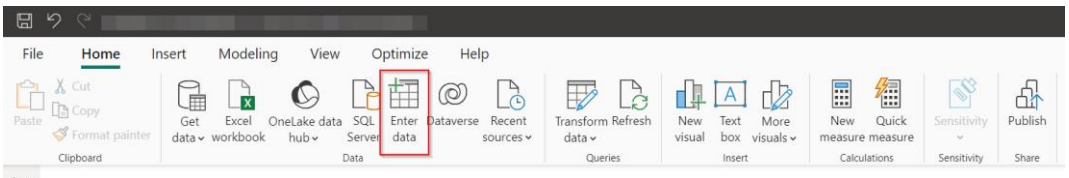
It is good practice to first create a **"Measures Table"**. This will be used to store all our measures in the same location.

1. First, setup the Canvas for Visualisation Development
   - Load the SimplePBIBackground file to the **Canvas Background**
   - Load the HRTheme.json file

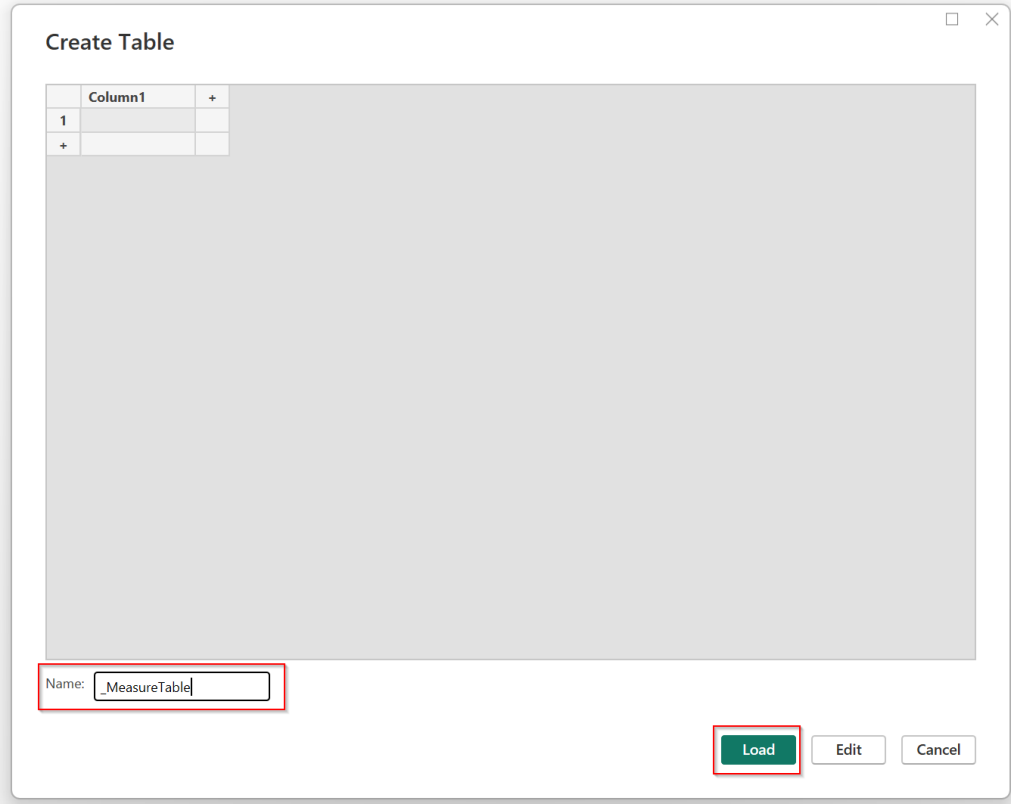


2. To create the Measure Table

   Navigate to the **Enter Data** icon at the top menu.



3. Change the Name of the Table to "**_MeasureTable**" and click **Load**.



This will create a table in the Data pane called **"_MeasureTable".** The underscore prefix ensures that this is the first table displayed in the Data pane.

We want to create the following Measures, which will be used in our visualisations.

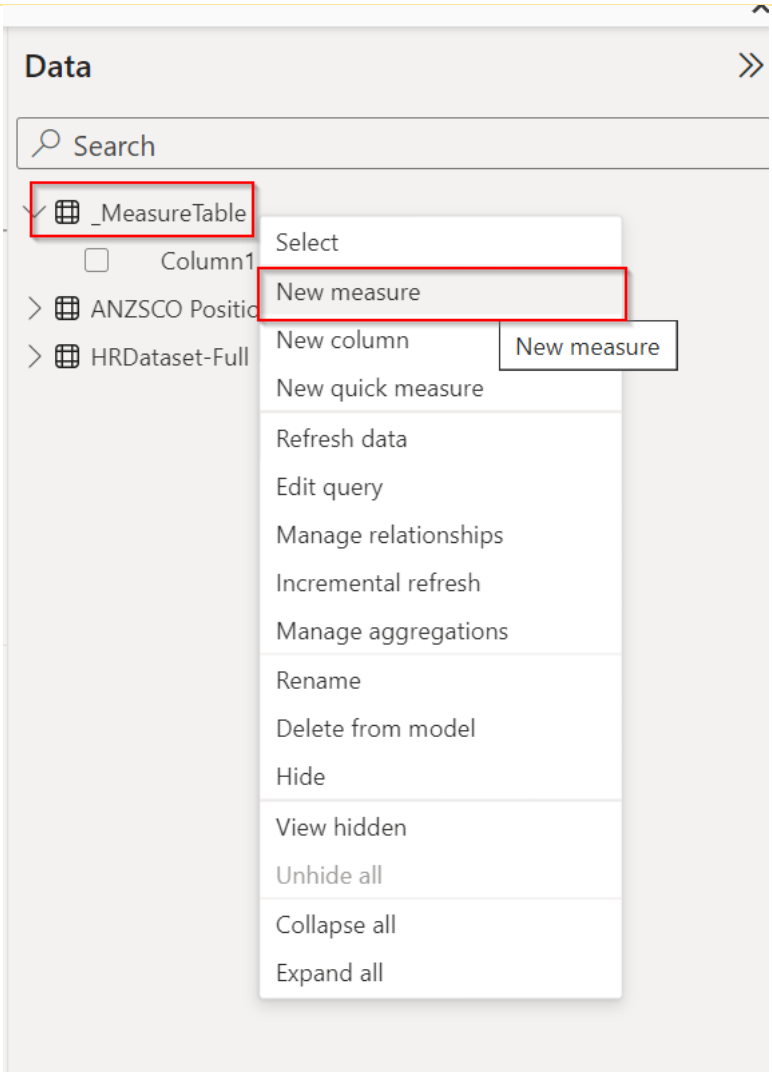
7. Count of Employees
8. Count of Employees (Casual)
9. Count of Employees (Fixed Term Full-Time)
10. Count of Employees (Fixed Term Part-Time)
11. Count of Employees (Full-Time)
12. Count of Employees (Part-Time)

We will be using a combination of the following DAX functions to achieve this:

- DISTINCTCOUNT()
- CALCULATE()

We will start with creating the **Count of Employees** measure.

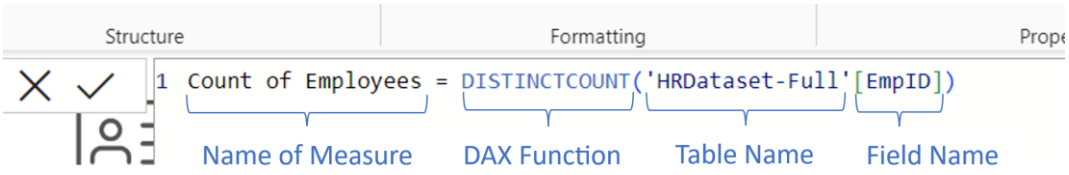
4. To create your first measure, right-click on the **_MeasureTable**, and select **New measure**



5. This will open the **Formula Editor**.

Enter the following text into the formula editor to create the **Count of Employees** measure.

```
Count of Employees = DISTINCTCOUNT('HRDataset-Full'[EmpID])
```



In DAX, you can also perform a DISTINCTCOUNT on employees based on a specific field. In our case, we want a DISTINCTCOUNT of our employees based on their [Employment Type]. This is achieved by using a combination of the **CALCULATE()** and **DISTINCTCOUNT()**

Let's create the **Count of Employees (Casual)** measure.

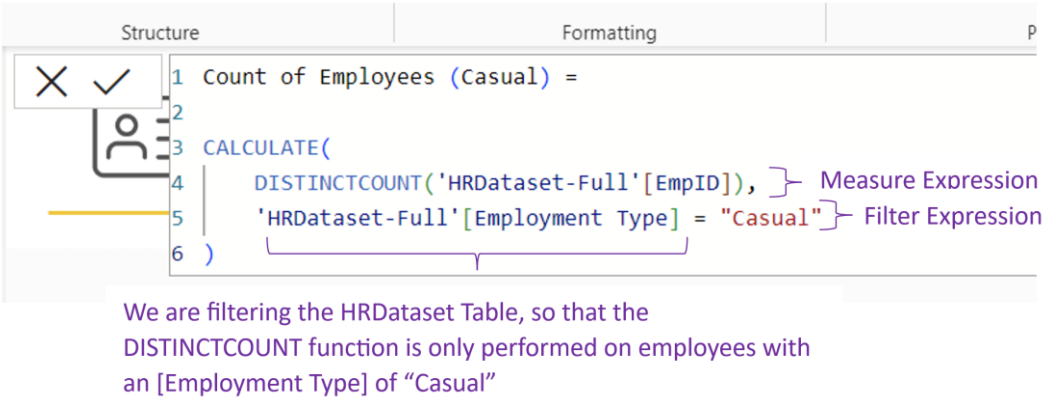
6. Right-click on the **_MeasureTable** and select **New measure.**

This will open the Formula Editor.

Enter the following text into the formula editor to create the **Count of Employees (Casual)** measure.

```
Count of Employees (Casual) =

CALCULATE(
    DISTINCTCOUNT('HRDataset-Full'[EmpID]),
    'HRDataset-Full'[Employment Type] = "Casual"
)
```



Note: You can add a line-break by pressing **Alt+Enter** while in the formula editor.

How it Works

The "Count of Employee (Casual)" measure is still performing a distinct count of employees, but is now only counting employees which have an [Employment Type] = "Casual".

- **CALCULATE**(): Allows you to perform a calculation while applying specific filters or conditions.
- **DISTINCTCOUNT**('HRDataset-Full'[EmpID]): This part of the formula counts the number of unique employee IDs (EmpID) in the 'HRDataset-Full' table.
- **Filter Condition**: 'HRDataset-Full'[Employment Type] = "Casual": This filter condition, used within the CALCULATE function, narrows down the data to include only records where the employment type is classified as "Casual."

7. Try and create Measures to count the other [Employment Type]. The filter expression is provided below:

   1. **Count of Employees (Fixed Term Full-Time)**
      `'HRDataset-Full'[Employment Type] = "Fixed Term Full-Time"`

   2. **Count of Employees (Fixed Term Part-Time)**
      `'HRDataset-Full'[Employment Type] = "Fixed Term Part-Time"`

   3. **Count of Employees (Full-Time)**
      `'HRDataset-Full'[Employment Type] = "Fixed Term Part-Time"`

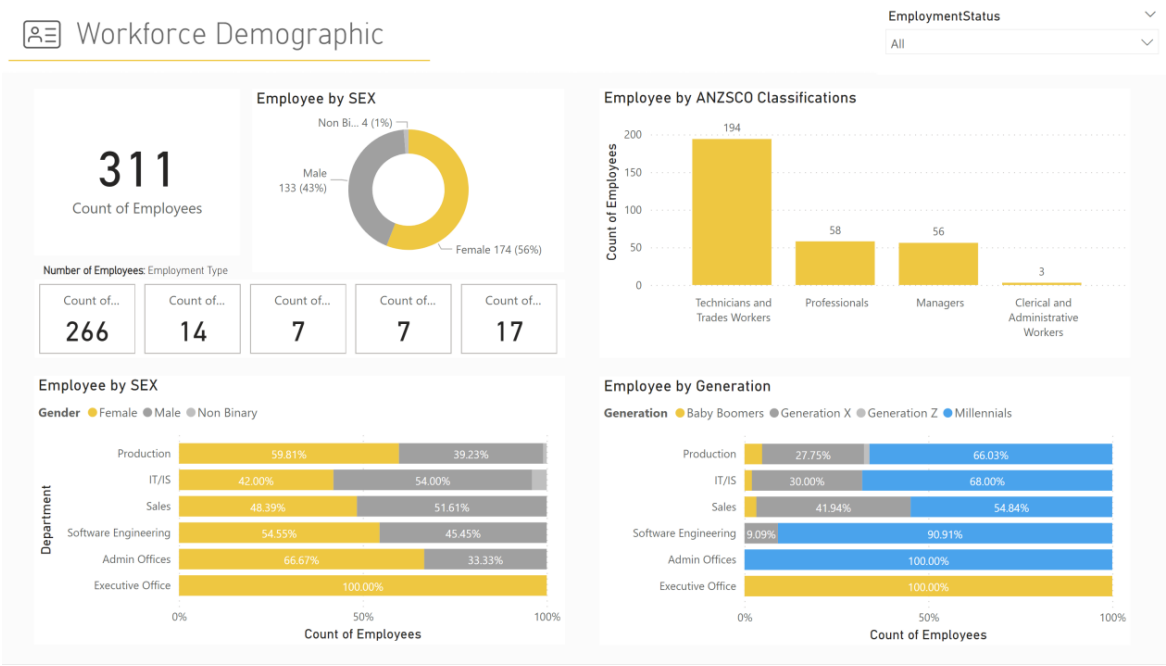
4. **Count of Employees (Part-Time)**

```
'HRDataset-Full'[Employment Type] = "Part-Time"
```

Part 4: Creating Visualisation using Measures

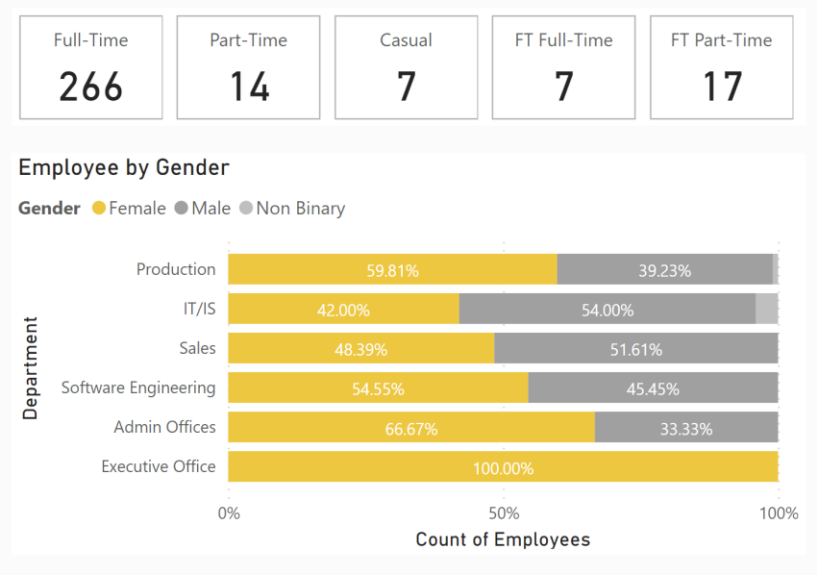
With the explicit measure's created, we should be able to create the following visualisation.

As demonstrated in Exercise 1, setup your report with the canvas background, icon, textbox header, and JSON theme.

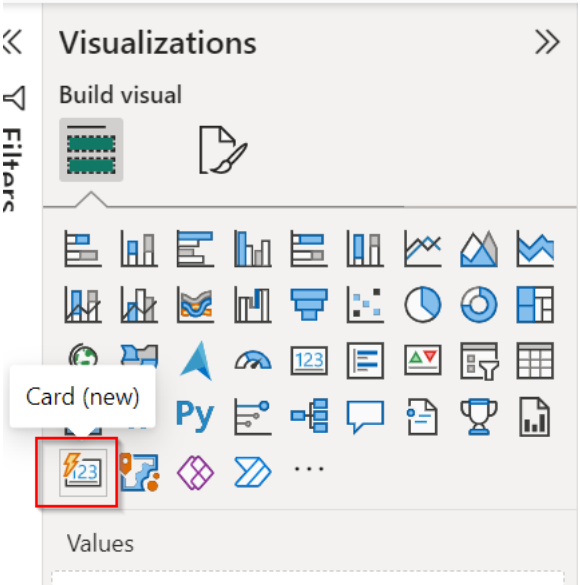


Let's start off by creating these two visualisations

1. **Card (new)** Visual
2. **100% Stacked Bar** Visual

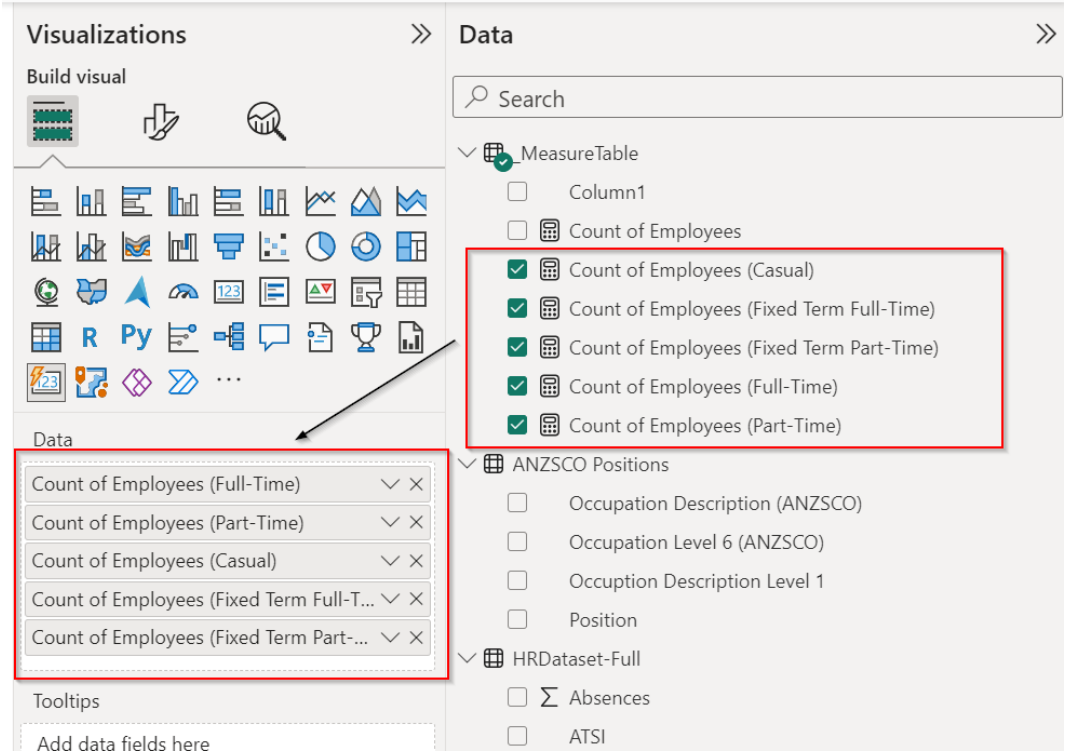


8. Select the **Card (new)** option from the Visualisation Pane.

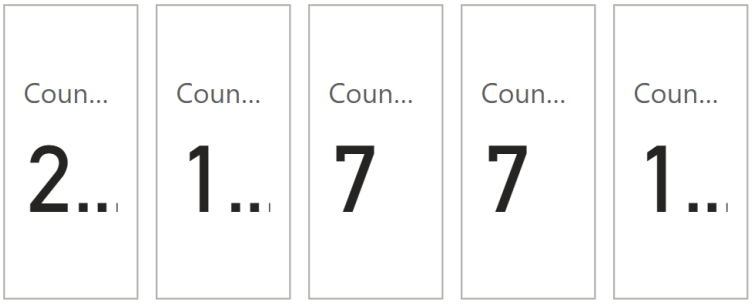


9. From the **Data** pane, navigate to **_MeasureTable**. Select or Drag the following fields to the Data Section located within the Visualisations Pane.

a) [Count of Employees (Full-Time)]
b) [Count of Employees (Part-Time)]
c) [Count of Employees (Casual)]
d) [Count of Employees (Fixed Term Full-Time)]
e) [Count of Employees (Fixed Term Part-Time)]



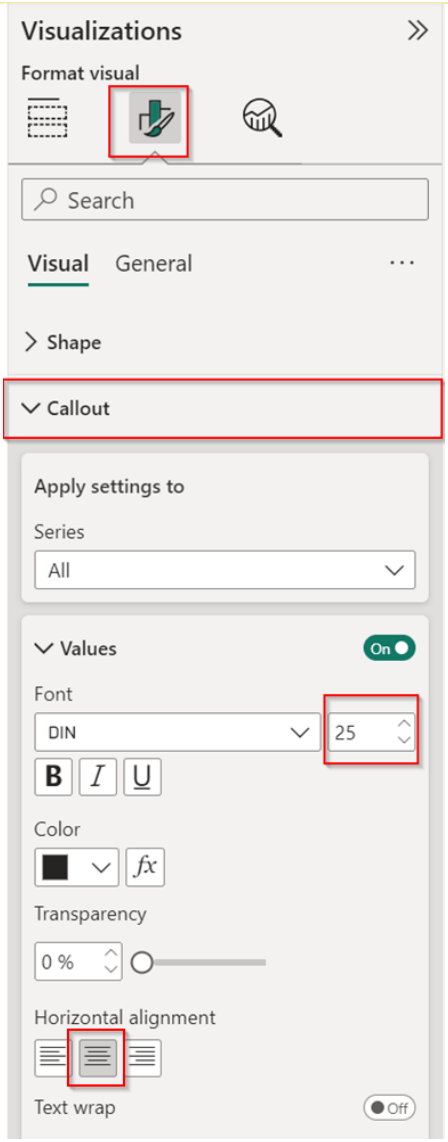
Check Point: Your Card visual will look similar to the one here.

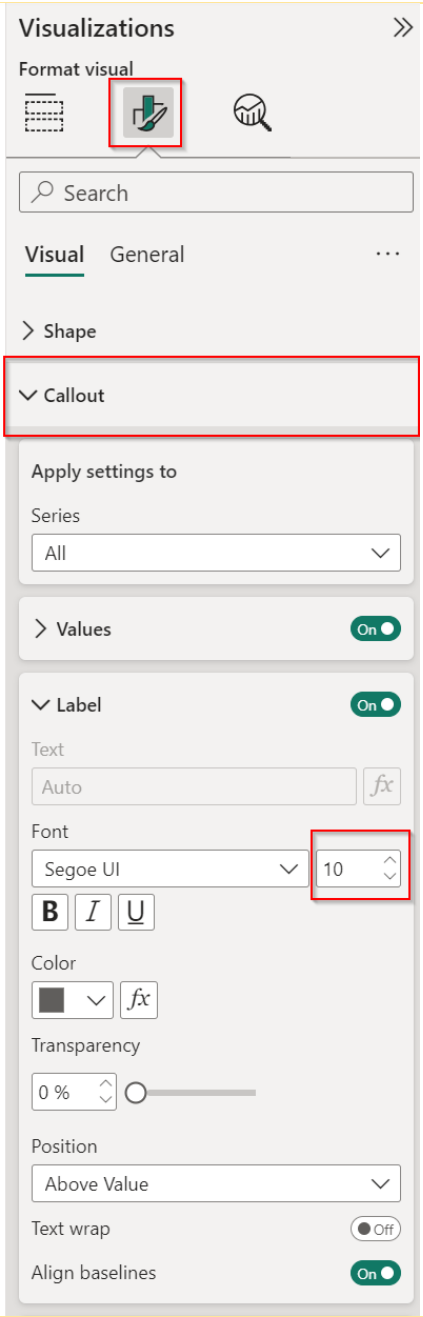


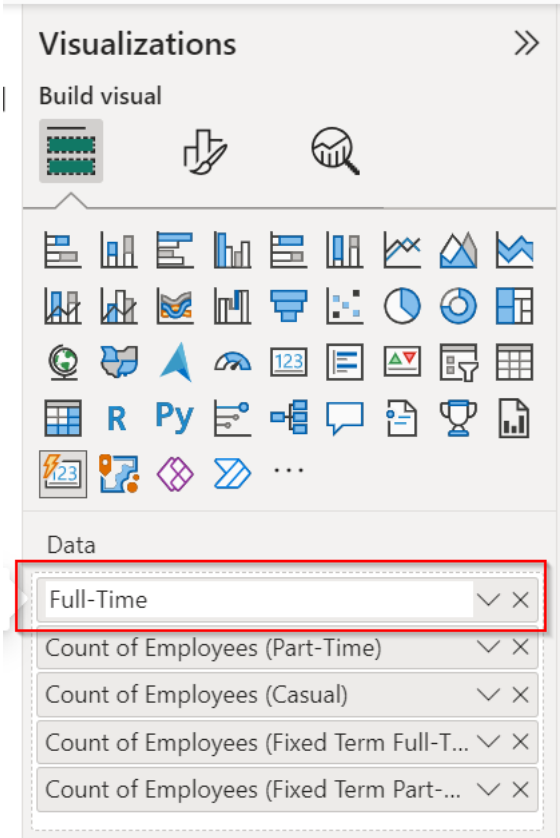
The size of the font is a little too big for our report. And the Labels for each Card could be more concise. Let's change the size of the font for the value and label, and then rename our labels.

10. Within the Visualisation Pane for the Card, navigate to the following area:
Format Visual - > Callout - > Values

Within the **Value** section, change the Font Size to **25**
Change **the Horizontal Alignment** to **Centre**


Visualizations
Format visual
Search
Visual
General
Shape
Callout
Apply settings to
Series
All
Values
On
Font
DIN
25
Color
Transparency
0 %
Horizontal alignment
Text wrap
Off

11.		Within the Visualisation Pane for the Card, navigate to the following area: **Format Visual - > Callout - > Label** Within the **Label** section, change the Font Size to **10**
12.		

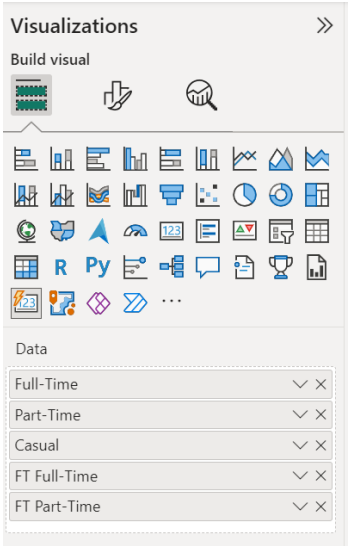


The Labels of the card can be re-labelled directly from Data Section of the Visualisation Pane.

Double click on the **[Count of Employees (Full-Time)]** field and rename it to **Full-Time**.

Hit **Enter** to confirm the change.

13. Try and change the remaining labels to the following:
    - Count of Employees (Part-Time) --> Part-Time
    - Count of Employees (Casual) --> Casual
    - Count of Employees (Fixed Term Full-Time) --> FT Full-Time
    - Count of Employees (Fixed Term Part-Time) --> FT Part-Time

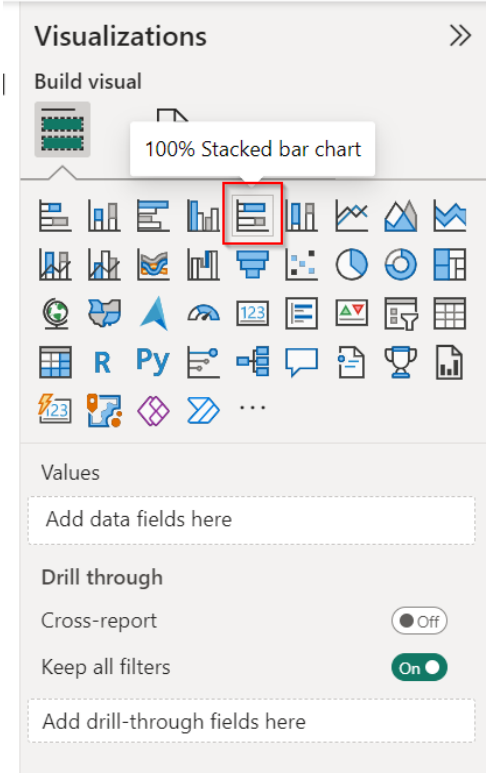


Check Point: Your Card visual will look similar to the one here after re-sizing.

Full-Time	Part-Time	Casual	FT Full-Time	FT Part-Time
266	14	7	7	17

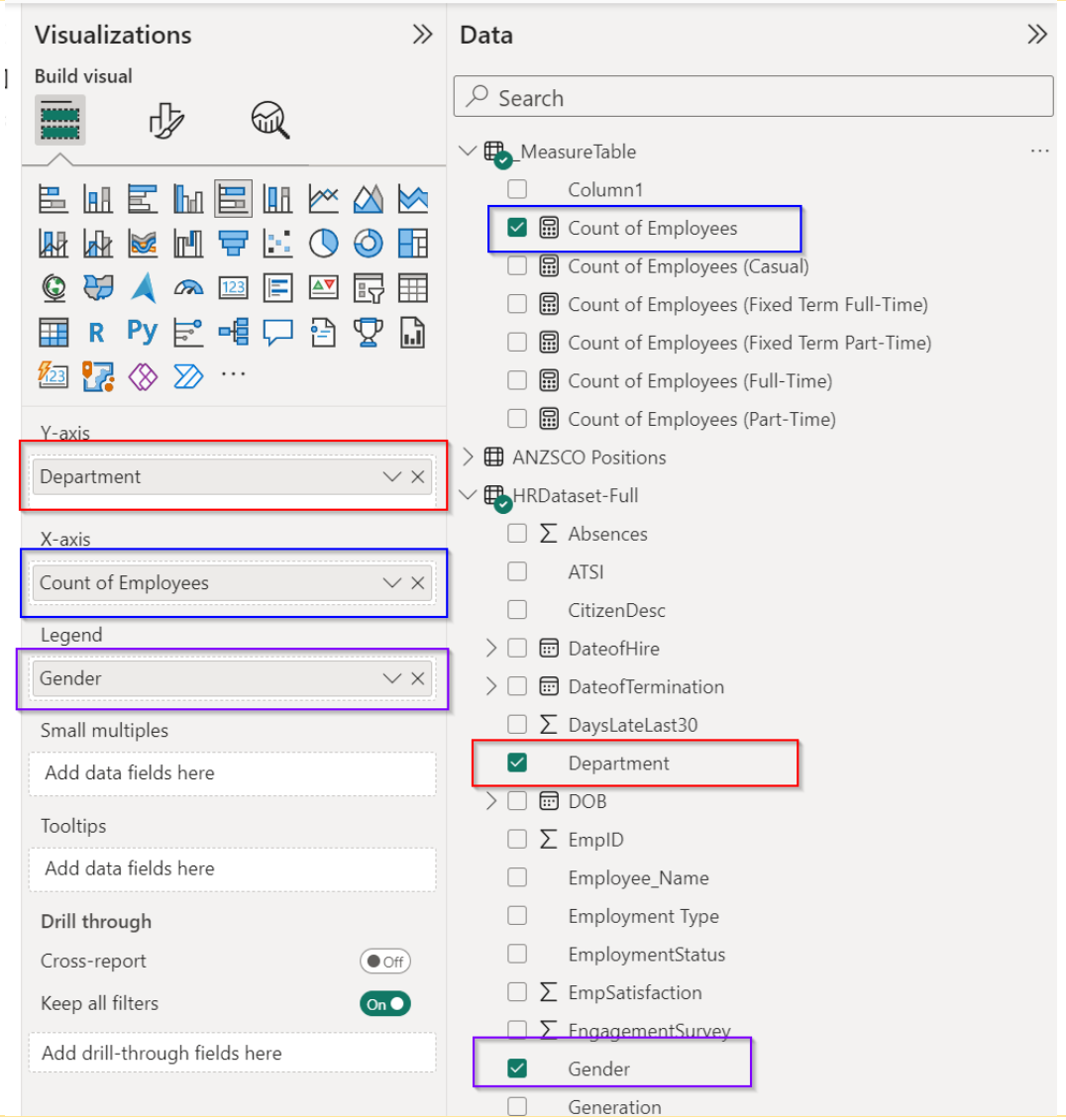
Now, let's create the 100% Stacked Bar Visual

14. Select the **100% Stacked Bar** visual option from the Visualisation Pane.

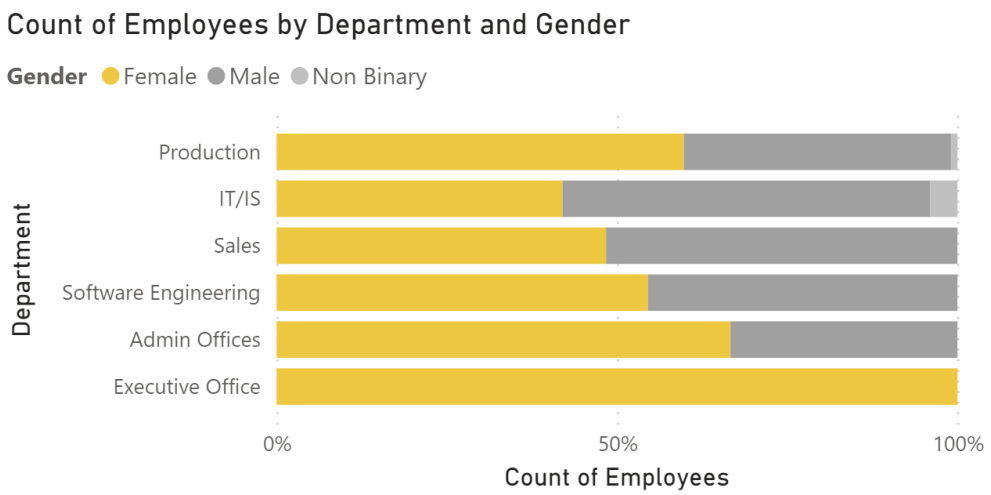


15. From the **Data** pane, navigate to **_MeasureTable** and **HRDataset-Full** tables. Select or Drag the following fields to the Data Section located within the Visualisations Pane.

- **Y-Axis:** [Department]
- **X-Axis:** [Count of Employees]
- **Legend**: [Gender]



Check Point: Your 100% Stacked Bar visual will should look like the one here after resizing.

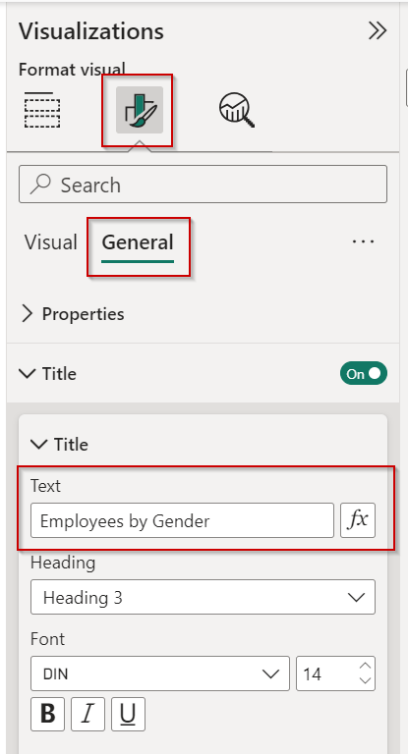


All that's remaining to replicate the original design is to change the Title and add data labels.

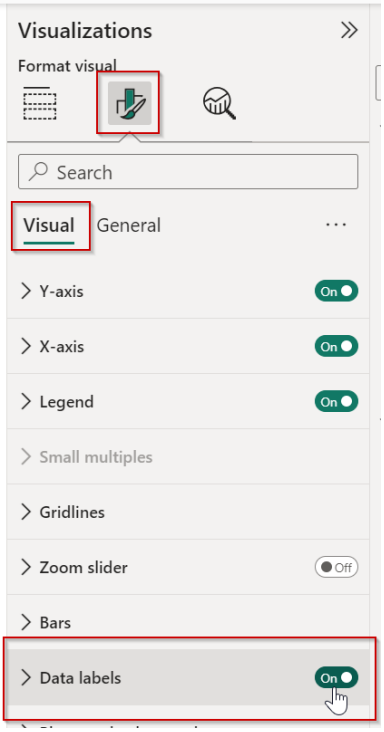
16. Within the Visualisation Pane for the Chart, navigate to the following area:

Format Visual - > General - > Title

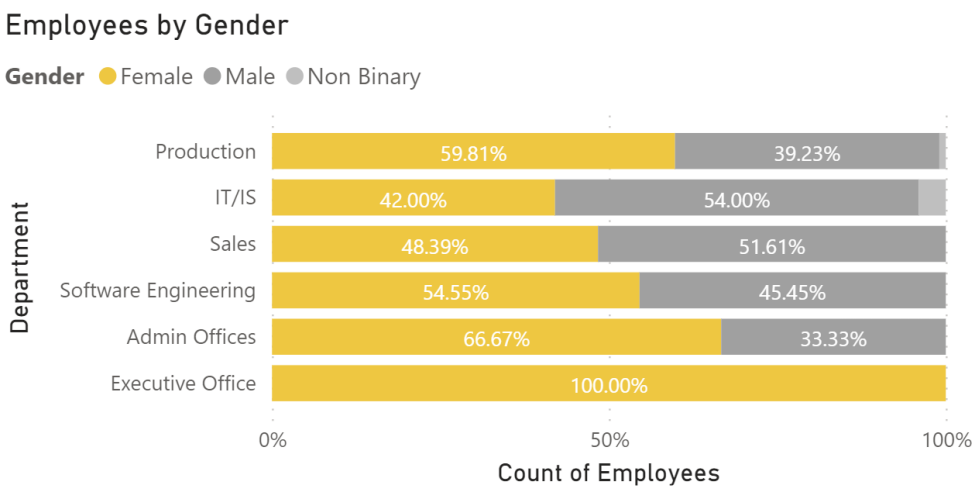
Within the **Title** section, change the Text to **Employees by Gender**



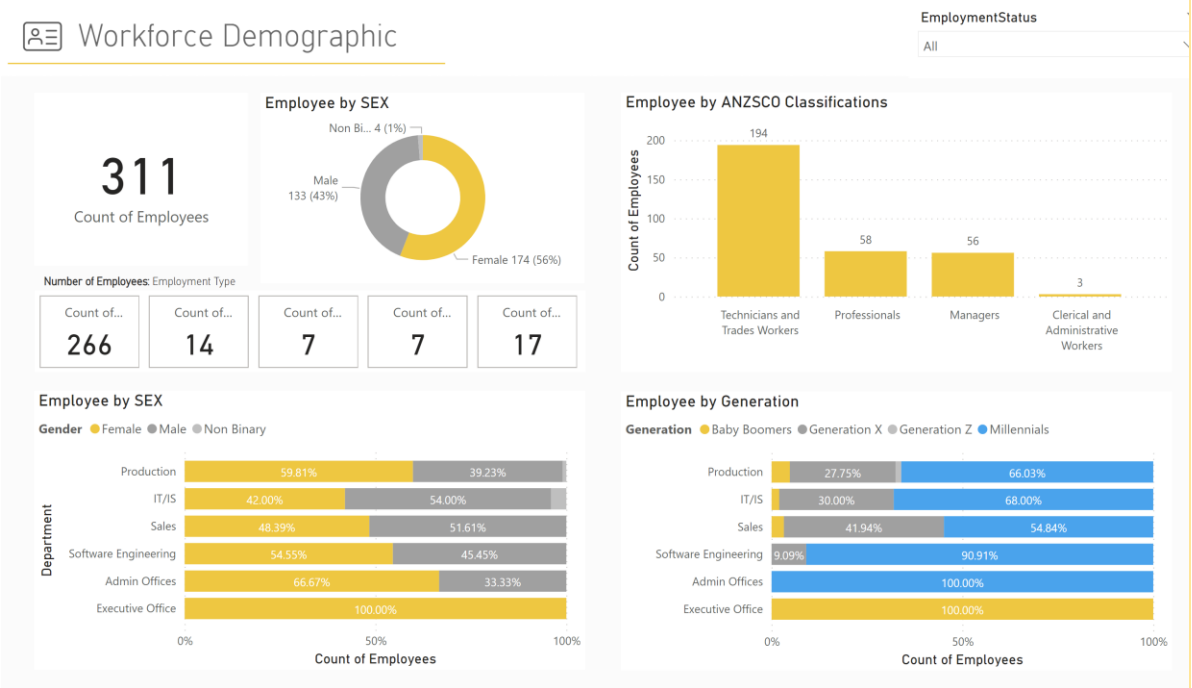
17. Within the Visualisation Pane for the Chart, navigate to the following area:
Format Visual - > Visual - > Data labels
Within the **Data labels** section, toggle data labels to **ON**



Check Point: Your 100% Stacked Bar visual will should look like the one here after resizing.



Try and create the remaining Visuals shown here:



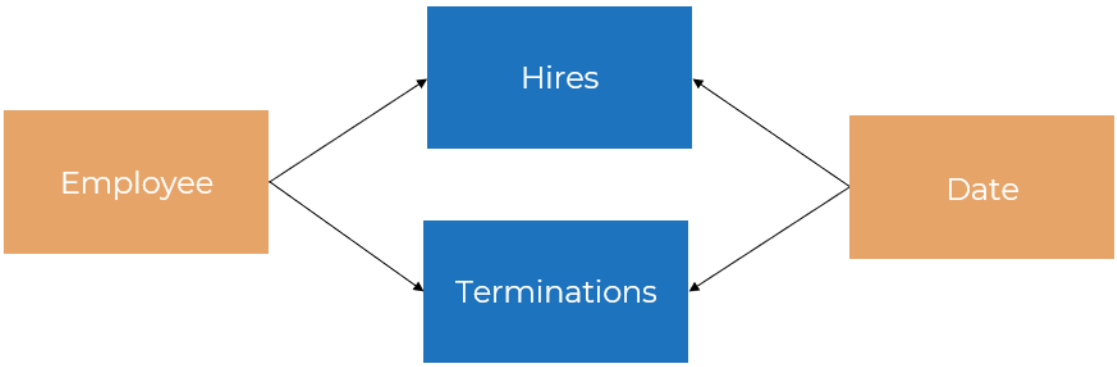
Note: The Top right bar chart, use the **[Occupation Description Level 1]** field from table **ANZSCO Positions** in the X-Axis section.

Exercise 3: Workforce Summary

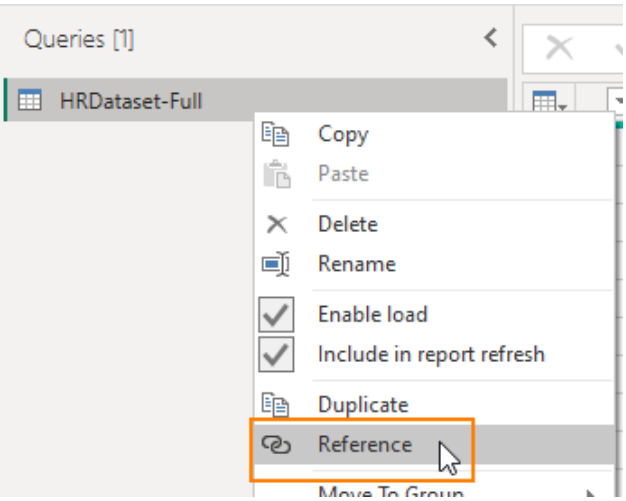
Learning Outcomes

- Design a basic data model
- Use Power Query to transform the flat file into a data model
- Create relationships in Power BI
- Create Simple Explicit Measures
- Create more visualisations using the new data model

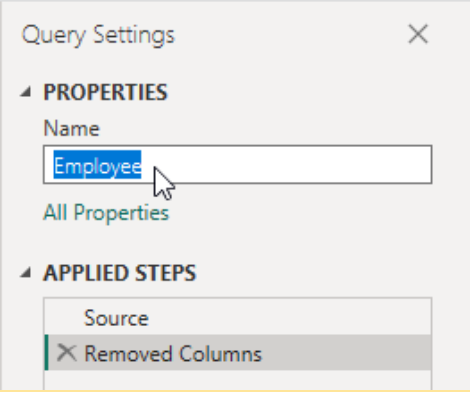
Reference Data Model



Part 1: Use Power Query to transform the flat file into the data model

First, we want to create the "Employee" dimension. This will contain all the columns that identify the employee, as well as any columns that are used to describe the employee.	
1.	Click on "Transform Data" to open the Power Query Editor
2.	Right click on "HRDataset-Full" from the Queries pane, and select "Reference"  This will create a new table called "HRDataset-Full (2)"

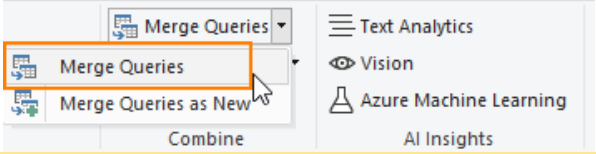
3. In the Query Settings pane on the right hand side, rename the table to be “Employee”



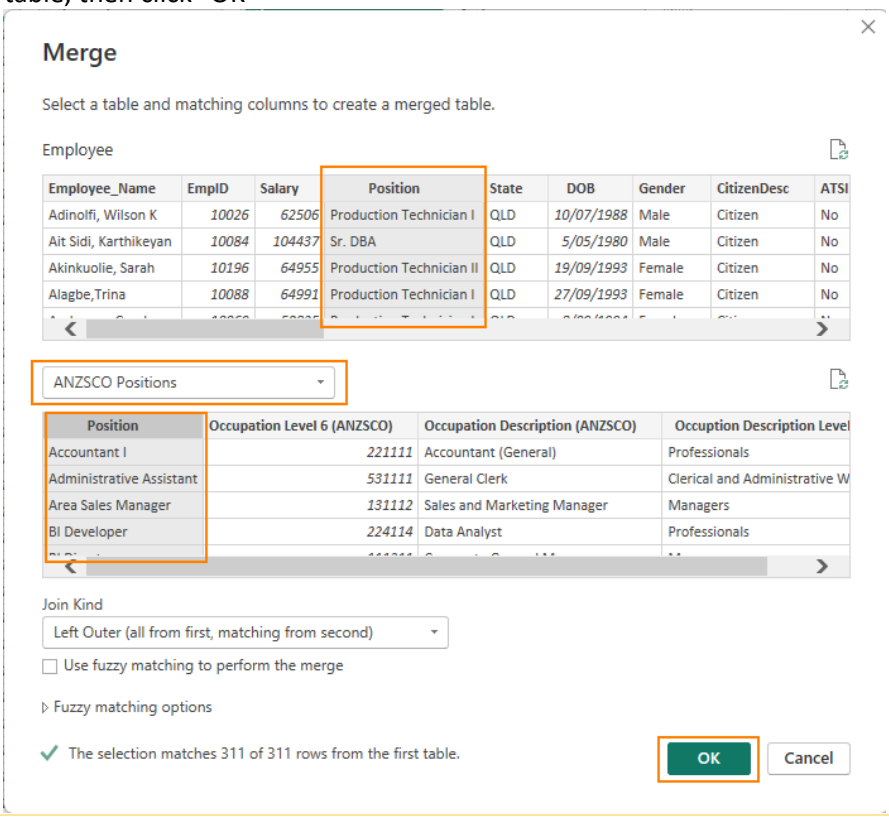
4. Remove the following columns from HRDataset-Full (2)
   - DateOfHire
   - DateOfTermination
   - TermReason
   - RecruitmentSource

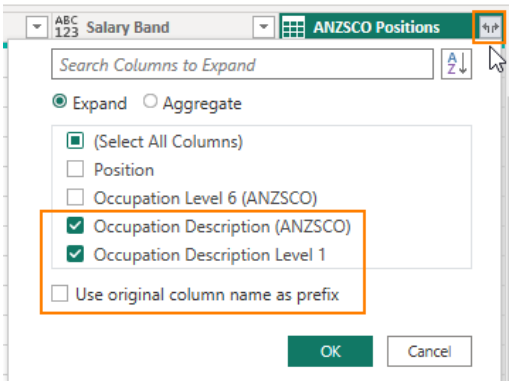
5. Since the ANZSCO Positions can be classified as employee attributes, we can combine the ANZSCO Position data into the Employee dimension.

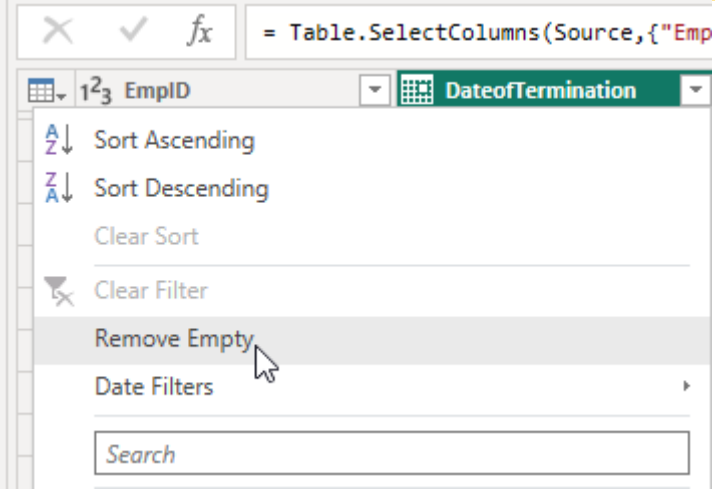
   In the Home ribbon, look for “Merge Query”, and click on “Merge Queries”



6. Click on “Position” on the top table
   Then choose “ANZSCO Positions” in the next box, and click on “Position” in the second table, then click “OK”



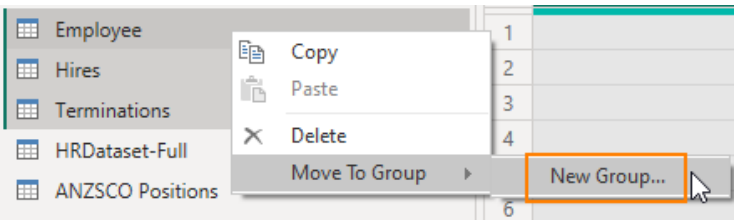
7.	Click on the expand button next to the newly created “ANZSCO Positions” column, then select the following two columns: • Occupation Description (ANZSCO) • Occupation Description (Level 1)  Then click OK – this will add the two columns into the Employees table.
Now we will create a simple Fact table for information about Hires	
8.	Right click on “HRDataset-Full” from the Queries pane, and select “Reference” This will create a new table called “HRDataset-Full (2)”
9.	Rename the query to be “Hires”
10.	Remove all columns except for the following using the “Remove Other Columns” option • EmpID • DateOfHire • RecruitmentSource
We will repeat a similar process for Terminations	
11.	Right click on “HRDataset-Full” from the Queries pane, and select “Reference” This will create a new table called “HRDataset-Full (2)”
12.	Rename the query to be “Terminations”
13.	Remove all columns except for the following using the “Remove Other Columns” option • EmpID • DateOfTermination • TermReason
14.	For the terminations table, we only want records where the employee has been terminated. Click on the arrow next to the “DateOfTermination” column, and click on “Remove Empty”



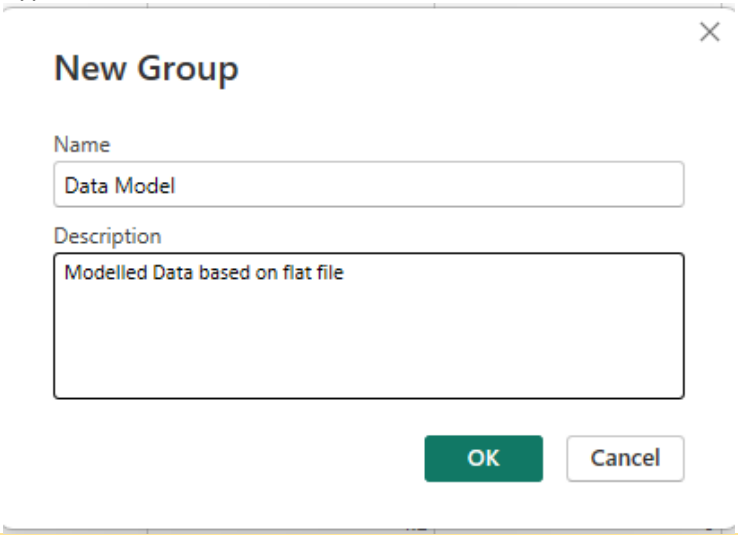
Now that we have both the source CSV file and the modelled tables in the Queries, we want to organise the tables a bit better so they are easier to find and work with.

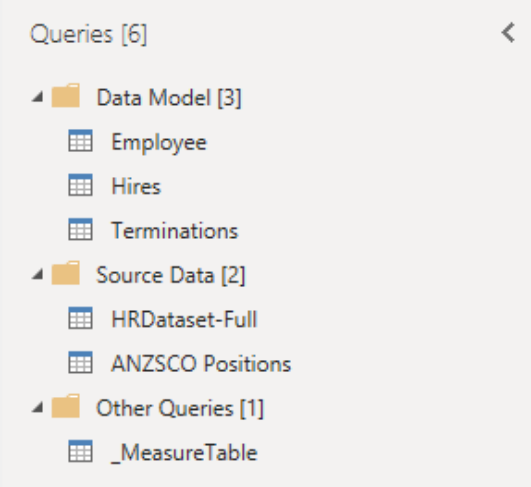
We want to create a folder called "Data Model" to house all our modelled data, as well as a folder called "Source Data" to house all our source data.

1. On the queries pane, select "Employee", "Hires" and "Terminations", then right click and select "Move To Group", then "New Group"

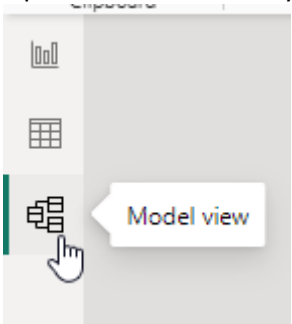


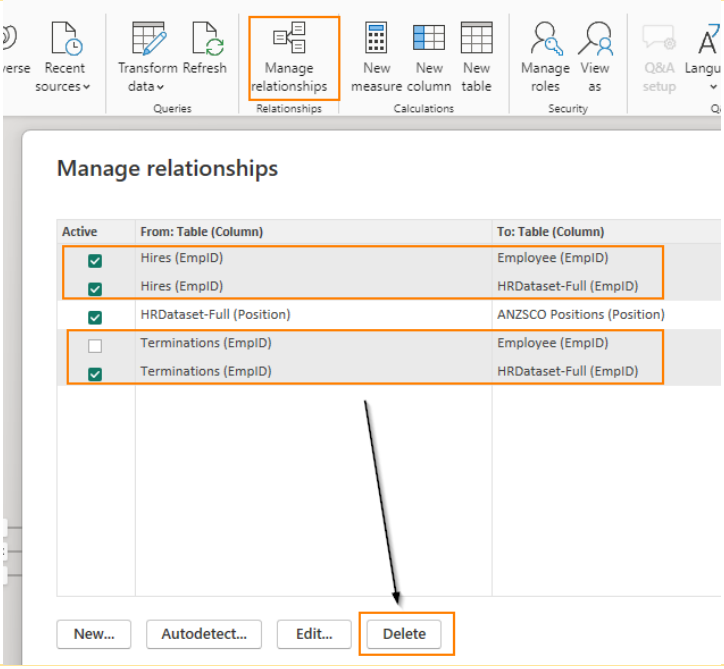
2. Type in "Data Model" for the name and click OK.



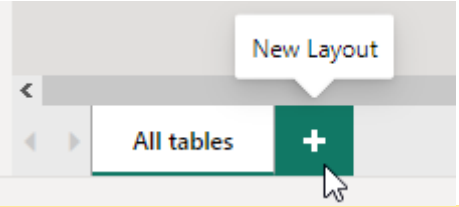
3.	Repeat for the “HRDataset-Full” and “ANZSCO Positions”, but add it into a group called “Source Data” . Your query pane should look like the following: 
4.	Click on “Close & Apply” and save your file.

Part 3: Create Relationships between the tables

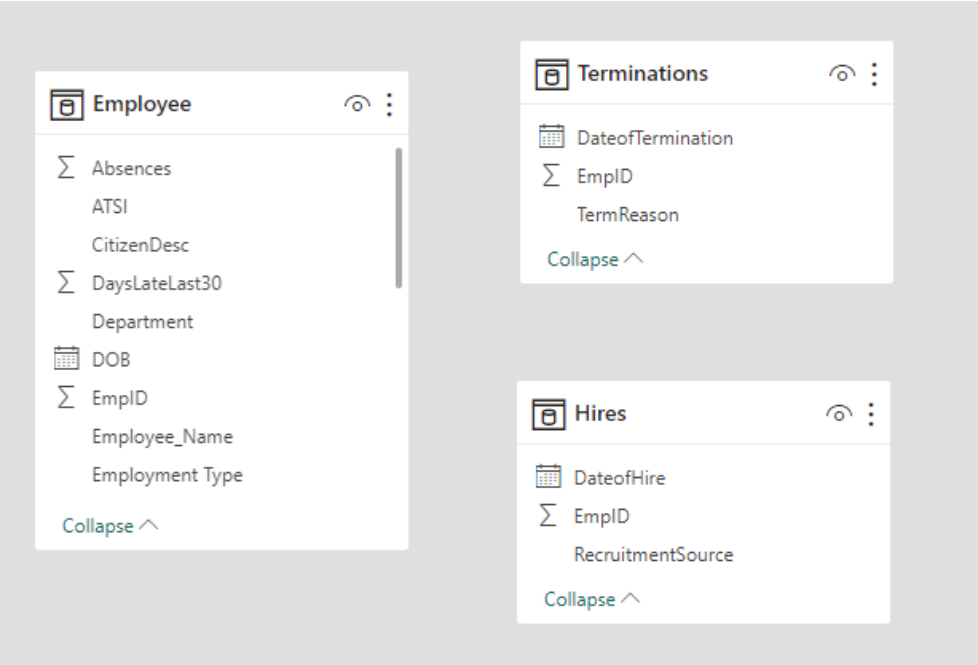
Power BI would have tried to create some relationships automatically based on the column names. In this case, we want to refine these relationships.	
1.	Open the Model view by clicking on the third icon on the left hand pane 
2.	Click on “Manage Relationships” on the top ribbon and delete the following auto created relationships



3. On the bottom, create a new layout by clicking on the + button, and name the new page "Data Model"

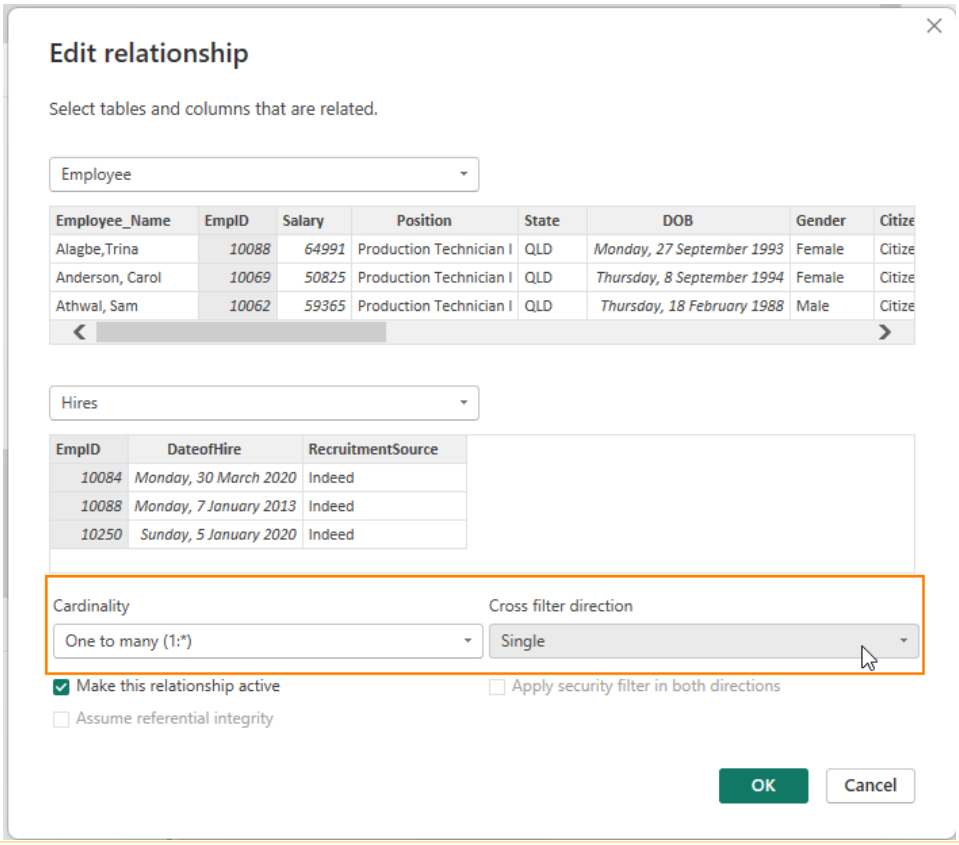


4. Drag in the following tables into the new layout:
   - Employee
   - Terminations
   - Hires

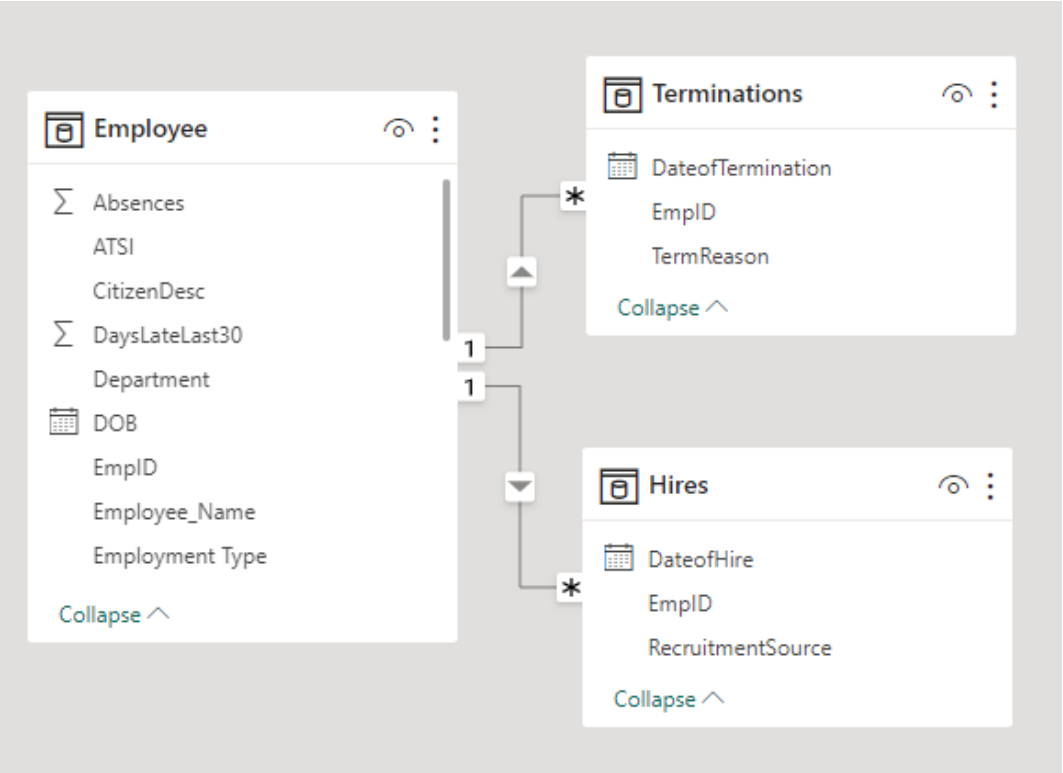


5. Click on EmpID in the Employee table and drag it to the EmpID in the Hires table.

Double click on the relationship created and set the following cardinality and cross filter direction, then click OK.



6. Repeat to connect Employee to Terminations. It should look like this:

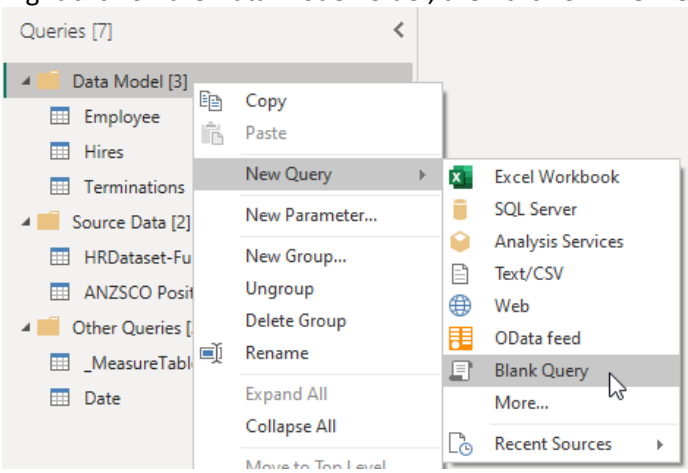


Part 4: Create and link Date Dimension

To be able to slice both the terminations and hires data by Date attributes, we will need to create a Date Dimension.

1. Click in “Transform Data” to open the Power Query Editor

2. Right click on the Data Model folder, then click on “New Query”, “Blank Query”



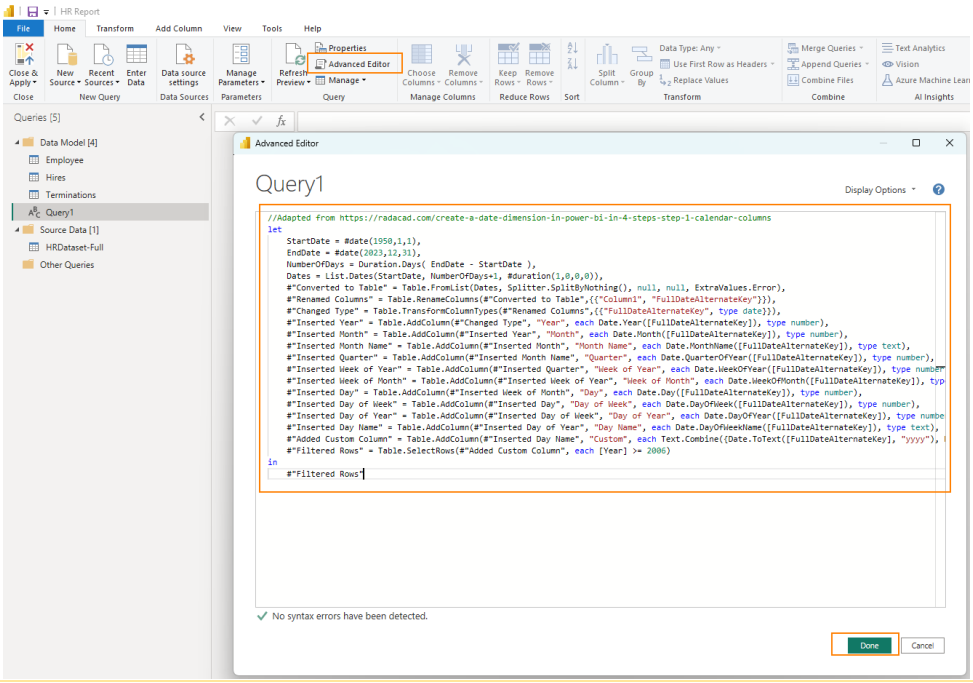
This will create a new table called “Query1”

3. In the Query Settings pane, rename the query to be “Date”

4. Click on “Advanced Editor”, and paste the following script, then click “Done”

```
//Adapted from https://radacad.com/create-a-date-dimension-in-power-bi-in-4-steps-step-1-calendar-columns
let
   StartDate = #date(1950,1,1),
   EndDate = #date(2023,12,31),
   NumberOfDays = Duration.Days( EndDate - StartDate ),
   Dates = List.Dates(StartDate, NumberOfDays+1, #duration(1,0,0,0)),
   #"Converted to Table" = Table.FromList(Dates, Splitter.SplitByNothing(), null, null, ExtraValues.Error),
   #"Renamed Columns" = Table.RenameColumns(#"Converted to Table",{{"Column1", "FullDateAlternateKey"}}),
   #"Changed Type" = Table.TransformColumnTypes(#"Renamed Columns",{{"FullDateAlternateKey", type date}}),
   #"Inserted Year" = Table.AddColumn(#"Changed Type", "Year", each Date.Year([FullDateAlternateKey]), type number),
   #"Inserted Month" = Table.AddColumn(#"Inserted Year", "Month", each Date.Month([FullDateAlternateKey]), type number),
   #"Inserted Month Name" = Table.AddColumn(#"Inserted Month", "Month Name", each Date.MonthName([FullDateAlternateKey]), type text),
   #"Inserted Quarter" = Table.AddColumn(#"Inserted Month Name", "Quarter", each Date.QuarterOfYear([FullDateAlternateKey]), type number),
   #"Inserted Week of Year" = Table.AddColumn(#"Inserted Quarter", "Week of Year", each Date.WeekOfYear([FullDateAlternateKey]), type number),
   #"Inserted Week of Month" = Table.AddColumn(#"Inserted Week of Year", "Week of Month", each Date.WeekOfMonth([FullDateAlternateKey]), type number),
   #"Inserted Day" = Table.AddColumn(#"Inserted Week of Month", "Day", each Date.Day([FullDateAlternateKey]), type number),
   #"Inserted Day of Week" = Table.AddColumn(#"Inserted Day", "Day of Week", each Date.DayOfWeek([FullDateAlternateKey]), type number),
   #"Inserted Day of Year" = Table.AddColumn(#"Inserted Day of Week", "Day of Year", each Date.DayOfYear([FullDateAlternateKey]), type number),
   #"Inserted Day Name" = Table.AddColumn(#"Inserted Day of Year", "Day Name", each Date.DayOfWeekName([FullDateAlternateKey]), type text),
   #"Added Custom Column" = Table.AddColumn(#"Inserted Day Name", "Custom", each Text.Combine({Date.ToText([FullDateAlternateKey], "yyyy"), Date.ToText([FullDateAlternateKey], "MM") })),
```

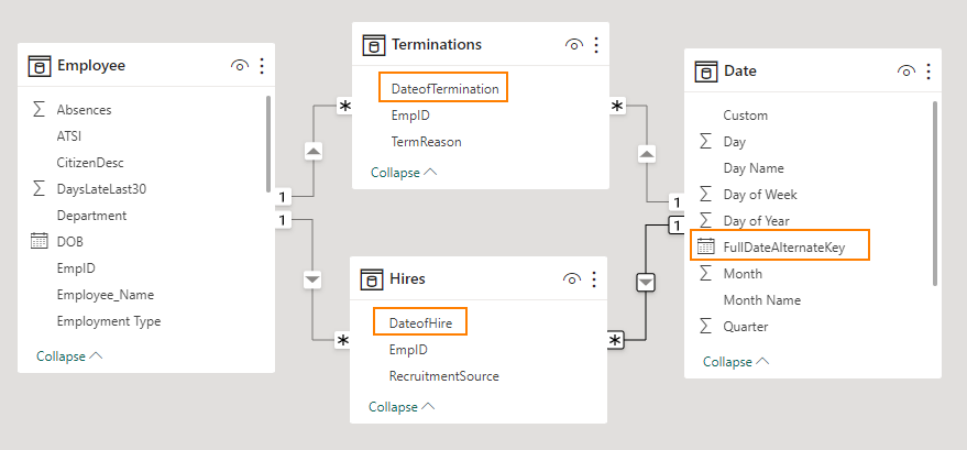
```
    #"Filtered Rows" = Table.SelectRows(#"Added Custom Column", each [Year] >= 2006)
in
    #"Filtered Rows"
```



5. Click “Close and Apply” to apply the changes
6. Go to the “Model View”, “Data Model” tab on the Power BI report
7. Add in the Date table to the canvas

   Drag the “FullDateAlternateKey” from the Date table, to the “DateOfTermination” from the Terminations table.

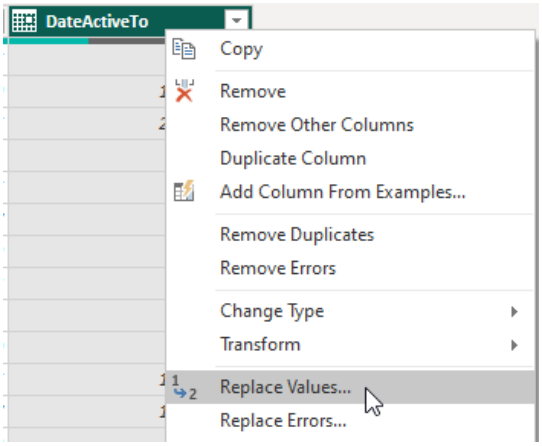
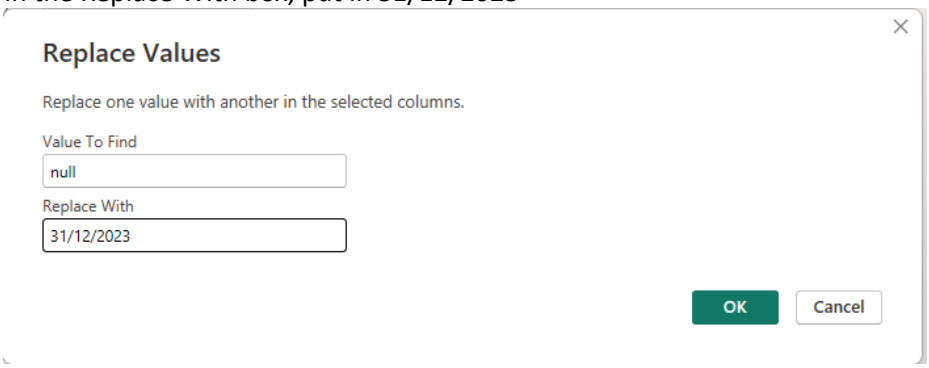
   Repeat with the “DateofHire” from the Hires table. Your model should now look like the following:



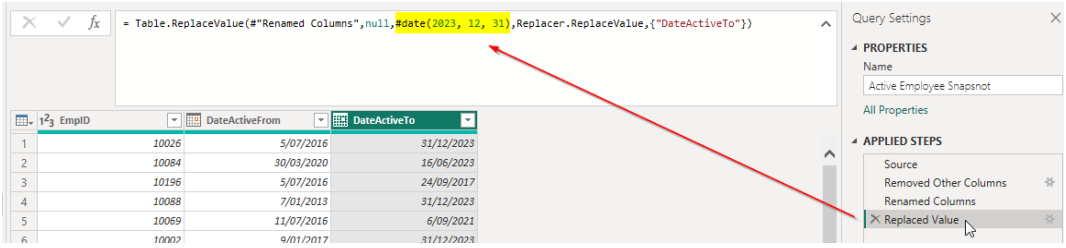
Part 5: Create Active Employee Snapshot

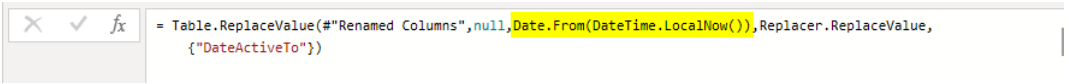
The Active Employee Snapshop is a slightly more complex table that will include employee ID and a range of active dates. This table is created by generating records for every date an employee was active in the organization, from their hire date until their termination date or until the current date if they are still employed.

This table will allow us to be able to do analytics on trend over time, head counts and lifecycle analysis.

1.	Click on “Transform Data” to open the Power Query Editor
2.	Right click on “HRDataset-Full” from the Queries pane, and select “Reference” This will create a new table called “HRDataset-Full (2)”
3.	In the Query Settings pane, rename the query to be “Active Employee Snapshot” Move this query to the Data Model group.
4.	Remove all columns except for: • EmpID • DateofHire • DateofTermination
5.	Rename the column: • DateofHire to DateActiveFrom • DateofTermination to DateActiveTo
6.	Right click on DateActiveTo and click on “Replace Values” 
7.	In the Value To Find box, put in “null” In the Replace With box, put in 31/12/2023  Click OK – you can know see the values that were null has now been replaced by 31/12/2023.
8.	To go one step further, we can put today’s date in “DateActiveTo” In the Query Settings pane, click on the “Replaced Value” step, then replace the text “#date(2023, 12, 31) with:

Date.From(DateTime.LocalNow())

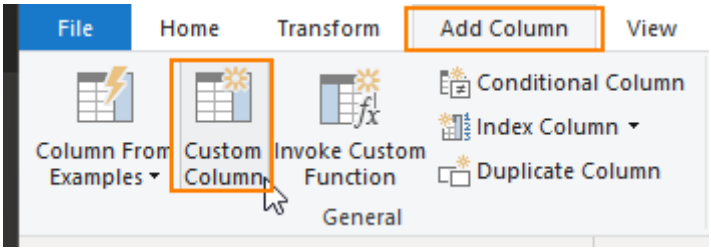




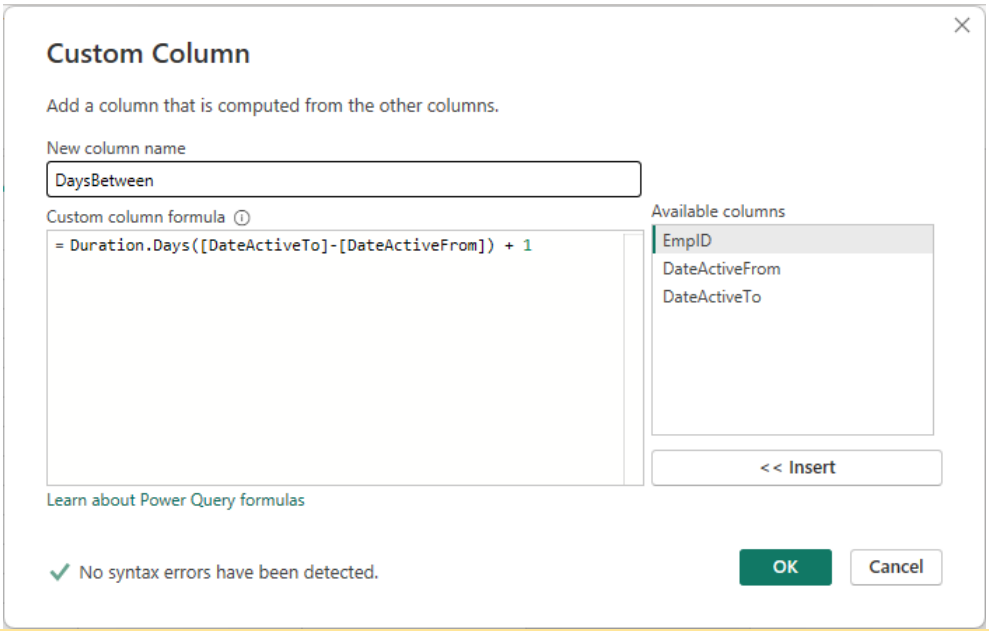
This will ensure that every time the dataset is refreshed, that it's using today's date

9. Next, we want to calculate how many days there are between DateActiveFrom and DateActiveTo

Under "Add Column", click on "Custom Column"



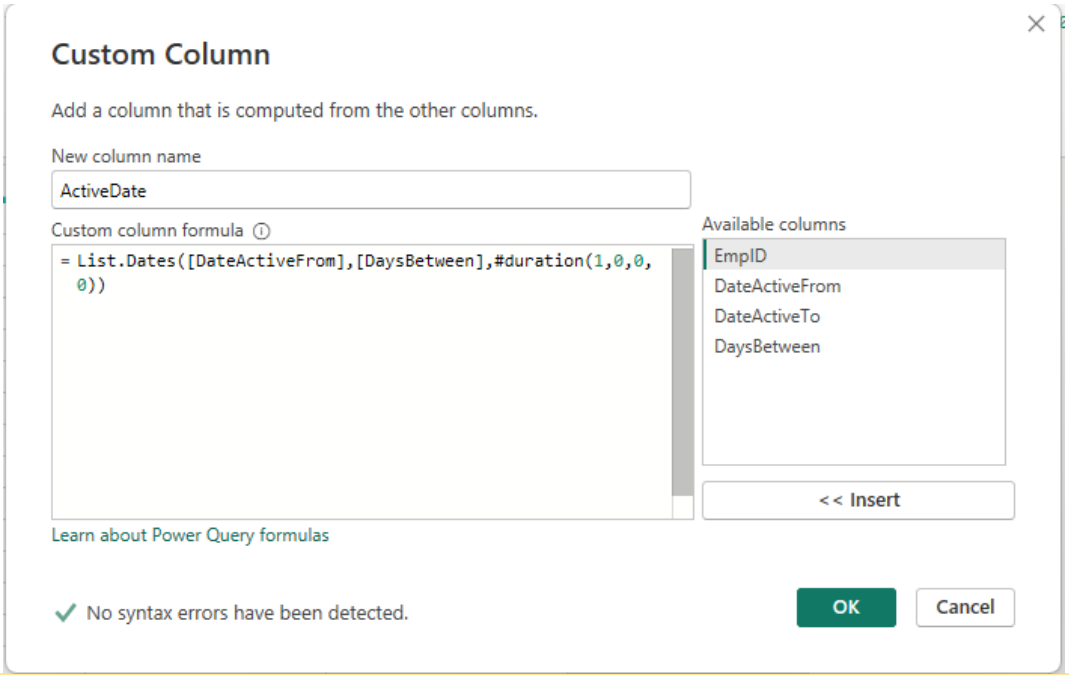
Enter the following:
New column name – "DaysBetween"
Custom column formula – "Duration.Days([DateActiveTo]-[DateActiveFrom]) + 1"
And click OK



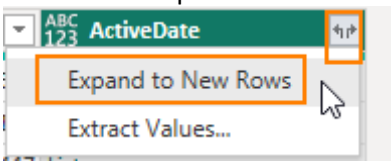
10. To create a List of dates between the DateActiveFrom and DateActiveTo

Under “Add Column”, click on “Custom Column”, and enter the following:

New column name – “ActiveDate”
Custom column formula –
“List.Dates([DateActiveFrom],[DaysBetween],#duration(1,0,0,0))”
And click OK



11. Click on the expand button to next to “ActiveDate” to expand to new rows:



12. Remove all columns except for:
    - EmpID
    - ActiveDate

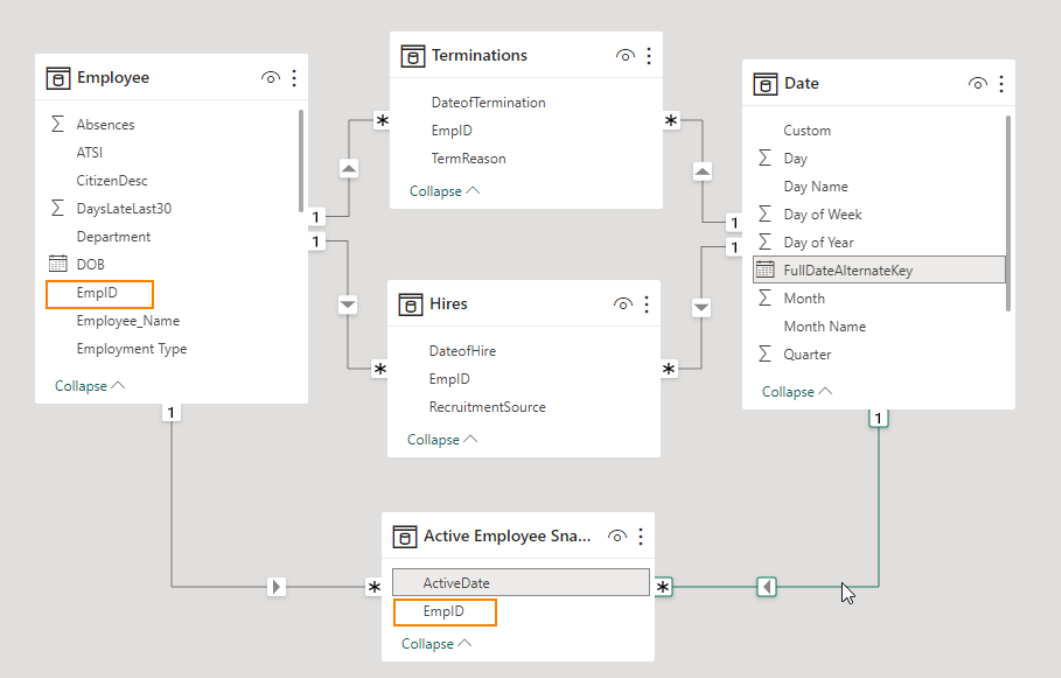
8. Click “Close and Apply” to apply the changes

9. Go to the “Model View” on the Power BI report

10. Add “Active Employee Snapshot” to the “Data Model” tab

    Drag the “EmpID” from the Employee table, to the “EmpID” from the Active Employee Snapshot table – this may have been done automatically.

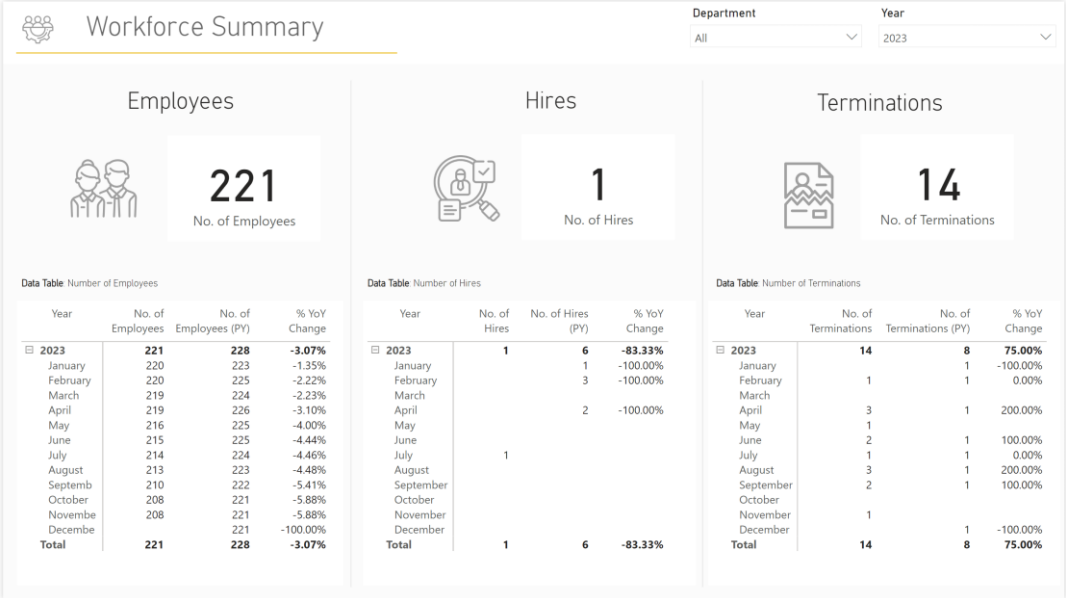
    Repeat with the “FullDateAlternateKey” from the Date table to the “ActiveDate” from the Active Employee Snapshot table.

	Your model should now look like the following: 
11.	Click on the "All Tables" view, and remove any auto relations created to the HRDataset-Full table
13.	Save your Power BI report.

Part 6: Create Measures

A well-designed data model can significantly reduce the effort required to produce a report by reducing the complexity and amount of DAX that needs to be written. Companies that high data maturity have established enterprise data models for their business users. This allows users to focus on generating insights and implementing plans to improve the business.

In this section, we will create a Workforce Summary report that give users an overview on the number of employees, hires, and termination within a department and year. The Data Model will allow us to slice across the Employee, Terminations, and Hires Fact Table using our DIM Tables (Date, and Employee)

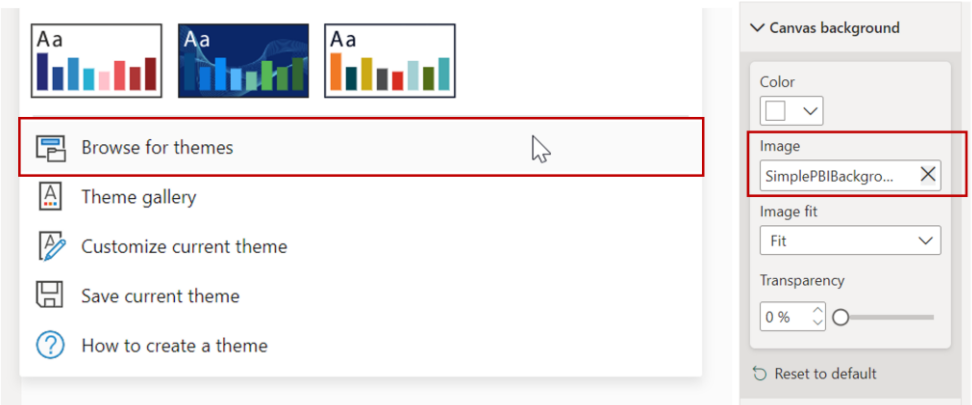


Year	No. of Employees	No. of Employees (PY)	% YoY Change
2023	221	228	-3.07%
January	220	223	-1.35%
February	220	225	-2.22%
March	219	224	-2.23%
April	219	226	-3.10%
May	216	225	-4.00%
June	215	225	-4.44%
July	214	224	-4.46%
August	213	223	-4.48%
Septemb	210	222	-5.41%
October	208	221	-5.88%
Novembe	208	221	-5.88%
Decembe		221	-100.00%
Total	221	228	-3.07%

Year	No. of Hires	No. of Hires (PY)	% YoY Change
2023	1	6	-83.33%
January		1	-100.00%
February		3	-100.00%
March			
April		2	-100.00%
May			
June			
July	1		
August			
September			
October			
November			
December			
Total	1	6	-83.33%

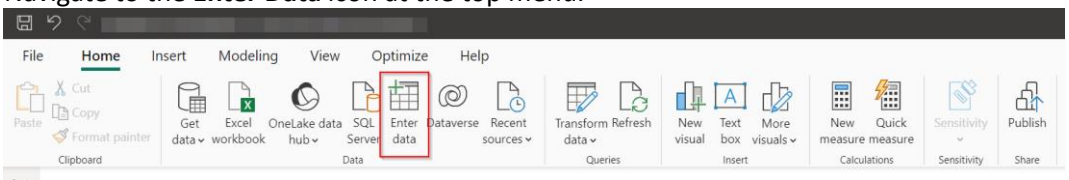
Year	No. of Terminations	No. of Terminations (PY)	% YoY Change
2023	14	8	75.00%
January		1	-100.00%
February	1	1	0.00%
March			
April	3	1	200.00%
May	1		
June	2	1	100.00%
July	1	1	0.00%
August	3	1	200.00%
September	2	1	100.00%
October			
November	1		
December		1	-100.00%
Total	14	8	75.00%

1. First, setup the Canvas for Visualisation Development
   - Load the SimplePBIBackground file to the **Canvas Background**
   - Load the HRTheme.json file

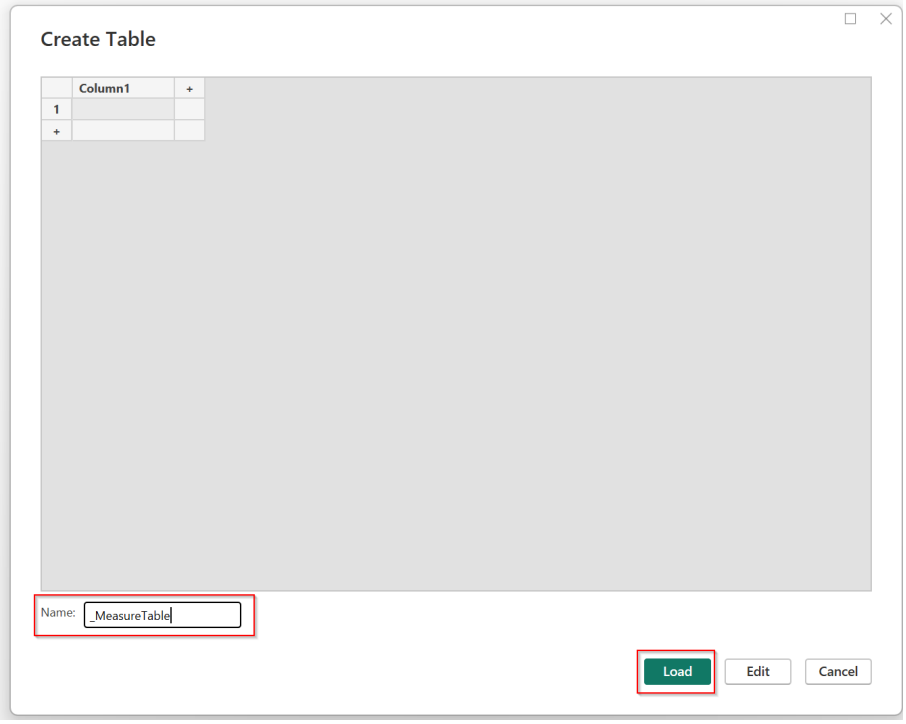


2. Next, create the Measure Table.

   Navigate to the **Enter Data** icon at the top menu.

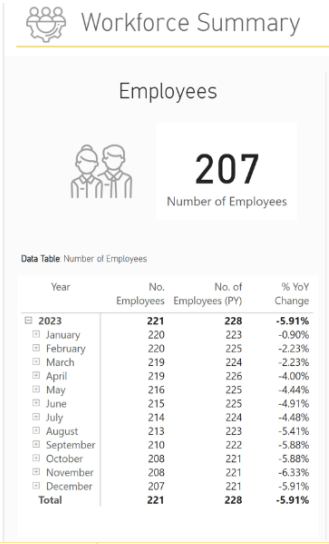


3. Change the Name of the Table to "**_MeasureTable**" and click **Load**.



This will create a table in the Data pane called **"_MeasureTable".** The underscore prefix ensures that this is the first table displayed in the Data pane.

Next, we'll create the measures required to re-produce this section of the report.



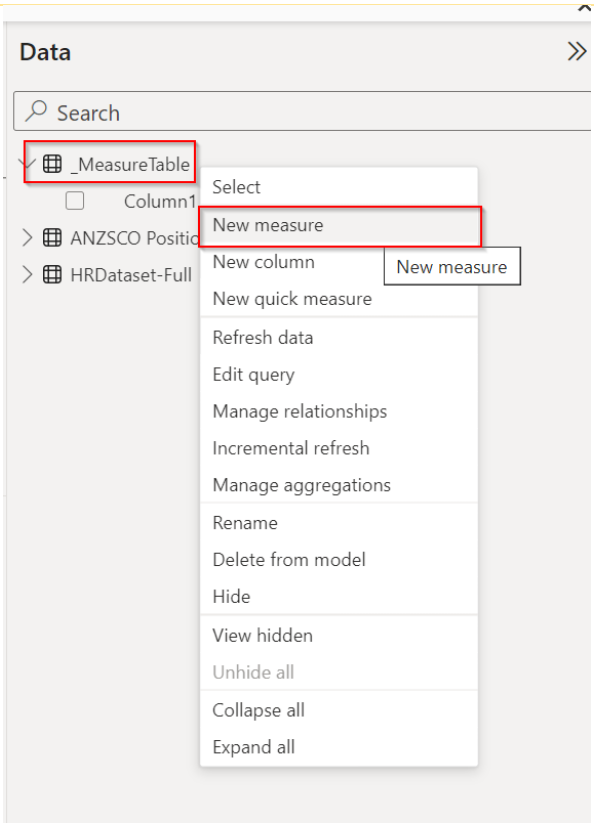
Year	No. Employees	No. of Employees (PY)	% YoY Change
2023	221	228	-5.91%
January	220	223	-0.90%
February	220	225	-2.23%
March	219	224	-2.23%
April	219	226	-4.00%
May	216	225	-4.44%
June	215	225	-4.91%
July	214	224	-4.48%
August	213	223	-5.41%
September	210	222	-5.88%
October	208	221	-5.88%
November	208	221	-6.33%
December	207	221	-5.91%
Total	221	228	-5.91%

1. No of Employees
2. No of Employees (PY)
3. No of Employees % YoY Change

We will be using a combination of the following DAX functions to achieve this:

- DISTINCTCOUNT()
- CALCULATE()
- DATEADD([Datefield], [Number of Intervals], [Interval])
- DIVIDE([numerator], [denominator])

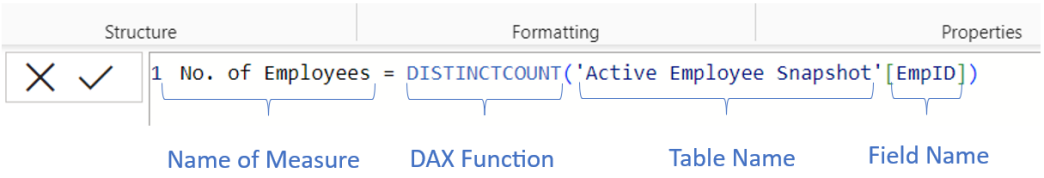
4. To create your first measure, right-click on the **_MeasureTable**, and select **New measure**



5. This will open the **Formula Editor**.

Enter the following text into the formula editor to create the **No. of Employees** measure.

```
No. of Employees = DISTINCTCOUNT('Active Employee Snapshot'[EmpID])
```



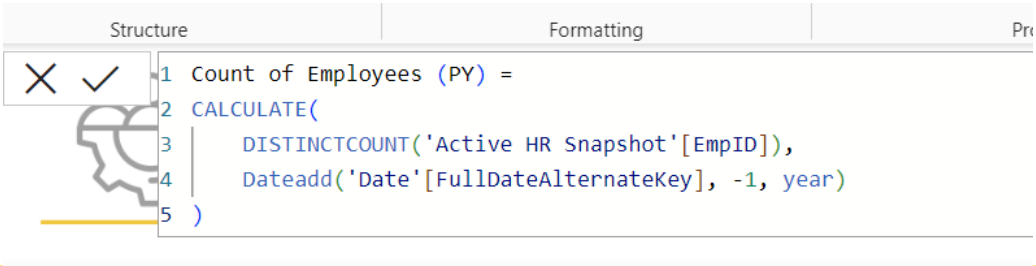
This will give us a distinct count of Employees from the [Active Employee Snapshot] table, based on the [EmpID] field.

6. Next, we want to create the **No. of Employees (PY)** measure, which counts the number of employees from the Previous Year.

Right-click on the **_MeasureTable** and select **New measure.**

Enter the following text into the formula editor to create the **No of Employees (PY)** measure.

```
Count of Employees (PY) =
CALCULATE(
    DISTINCTCOUNT('Active HR Snapshot'[EmpID]),
    Dateadd('Date'[FullDateAlternateKey], -1, year)
)
```



How it Works

The No. of Employee (PY) measure is calculating the total number of unique employees from the previous year. It does this by counting each employee ID once, and applying a time filter that shifts the focus of the calculation from the current year to the same period in the previous year.

The main change between this measure and the **No. of Employees** measure, is that we use the CALCULATE and DATAADD function to shift the time-period of the calculation back 1 year.

- **CALCULATE(): Allows you to perform a calculation while applying specific filters or conditions.**

- **DISTINCTCOUNT('Active Employee Snapshot'[EmpID]): This part of the formula counts the number of unique employee IDs (EmpID) in the [Active HR Snapshot] table.**

- **DATEADD('Date'[FullDateAlternateKey], -1, YEAR):** This is the filter condition within CALCULATE. DATEADD is a function that shifts a date by a specified number of intervals. In this case, it's shifting the dates in the Date[FullDateAlternateKey] column backward by one year (-1), effectively moving from the current year to the previous year.

7. Now, we can create the **No of Employees % YoY Change** measure, using the DIVIDE function, and the No. of Employee and No. of Employee (PY) measures.

   Right-click on the **_MeasureTable** and select **New measure.**

   Enter the following text into the formula editor to create the **No. of Employees % YoY Change** measure.

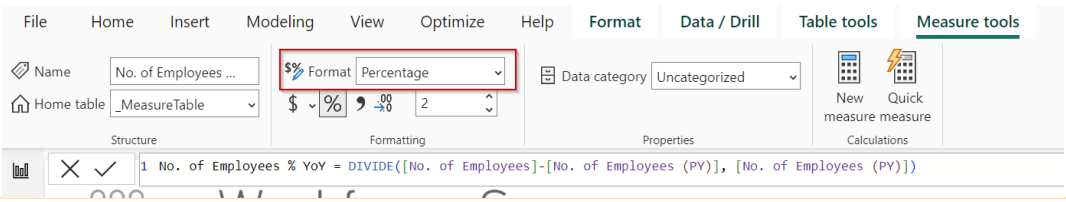
```
No. of Employees % YoY = DIVIDE([No. of Employees]-[No. of Employees
(PY)], [No. of Employees (PY)])
```

   Note: YoY = Year on Year

8. With the No. of Employees % YoY measure selected.

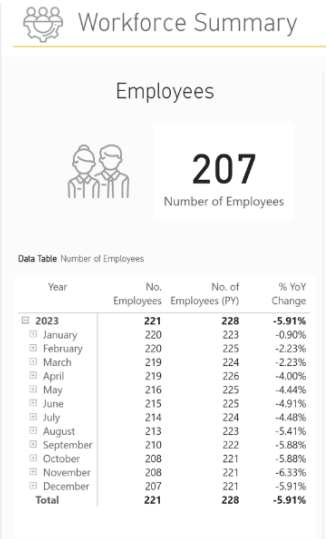
   Navigate to **Formatting** in the top menu.

   Change the **Format** to **Percentage**.



Part 6: Creating More Visualisations

Now that the measures are created, we can re-produce the following section of the report.



Year	No. Employees	No. of Employees (PY)	% YoY Change
2023	221	228	-5.91%
January	220	223	-0.90%
February	220	225	-2.23%
March	219	224	-2.23%
April	219	226	-4.00%
May	216	225	-4.44%
June	215	225	-4.91%
July	214	224	-4.48%
August	213	223	-5.41%
September	210	222	-5.88%
October	208	221	-5.88%
November	208	221	-6.33%
December	207	221	-5.91%
Total	221	228	-5.91%

This report will make use of the following core components:

1. Elements: to enhance the layout.

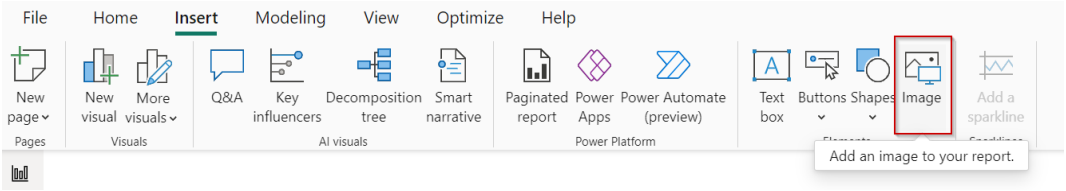
- Images
- Text Box

2. Card Visual: to provide a summary on the number of employees**.**

3. Matrix Visual: to provide a monthly breakdown on the number of employees and the year on year % change.

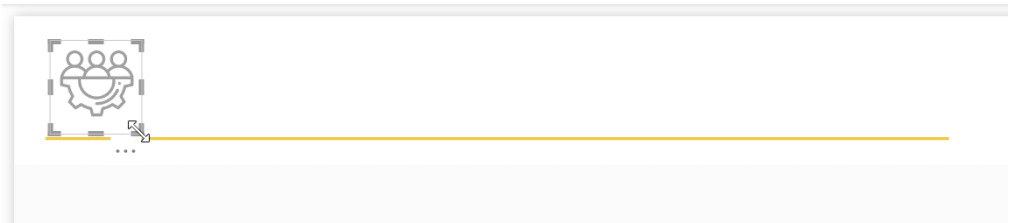
1. First, we will import the elements we need to reproduce the visual.

Navigate to **Elements** -> **Image** from the top menu.



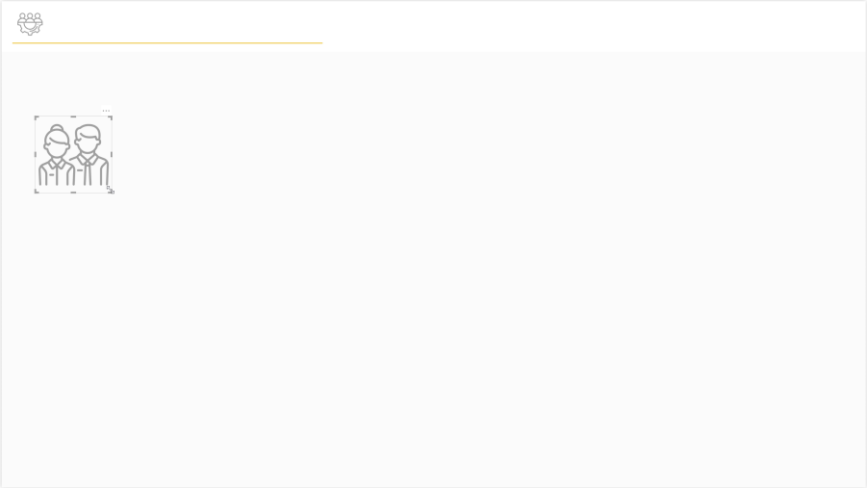
2. Import the **work.png** file from the Exercise 3 folder.

Resize the image and position it to the top left-hand corner of the header.



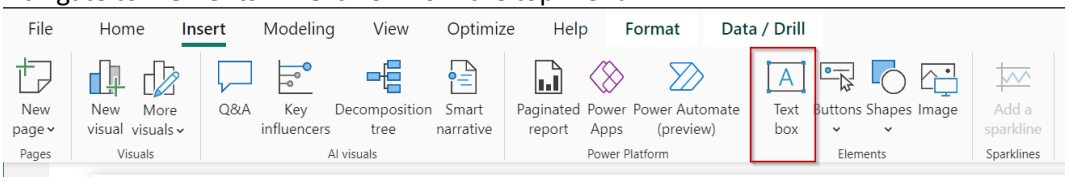
3. Import the **employees.png** file from the Exercise 3 folder.

Resize the image, and position it as shown here:

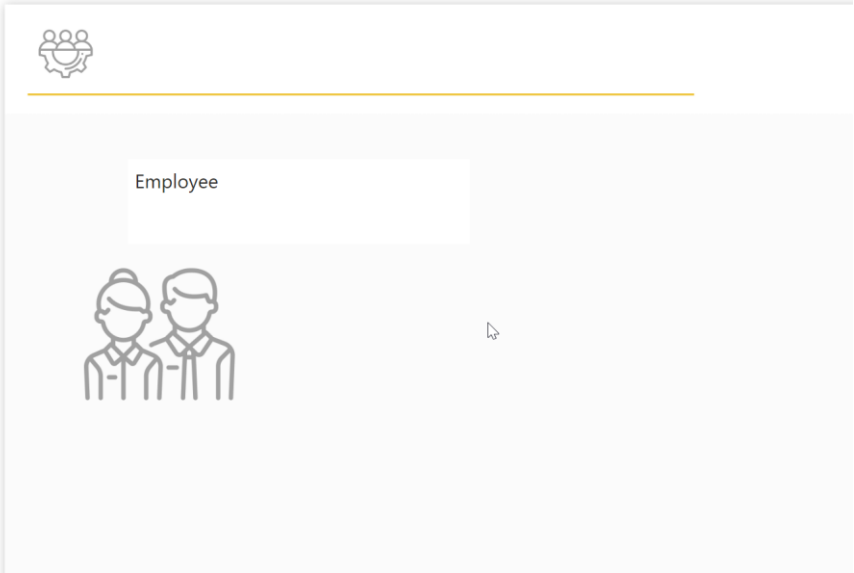


4. Next, we will add the text box.

Navigate to **Elements** -> **Text Box** from the top menu.

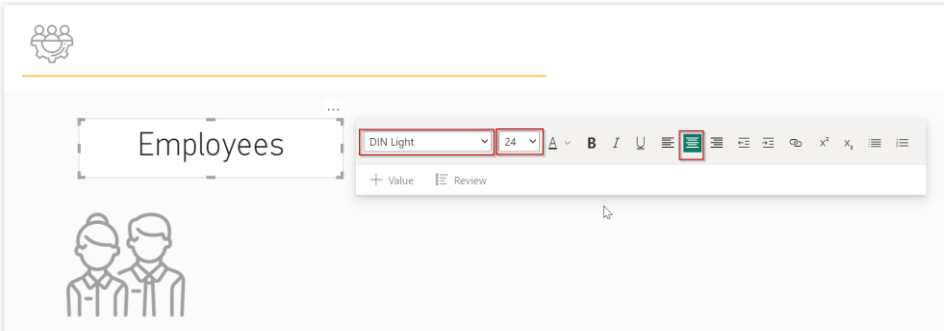


Enter “Employee” into the Textbox, and resize and position the Textbox as shown here.

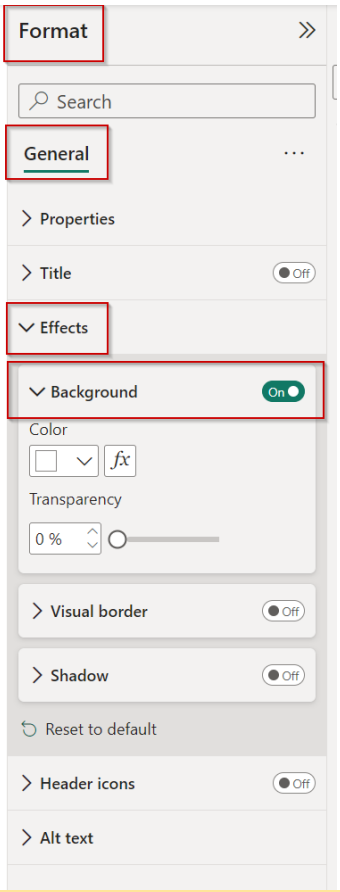


5. Update the Text Style to the following:

Font: Din Light
Font Size: 24
Text Alignment: Centre



6.

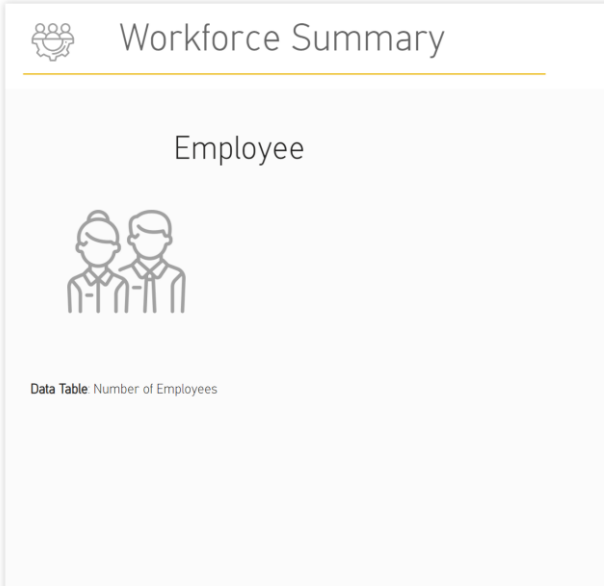


Next, remove the background white fill of the Textbox.

Navigate to **Format Pane** - > **General** - > **Effects** - > **Background**.

Toggle Background to **OFF**

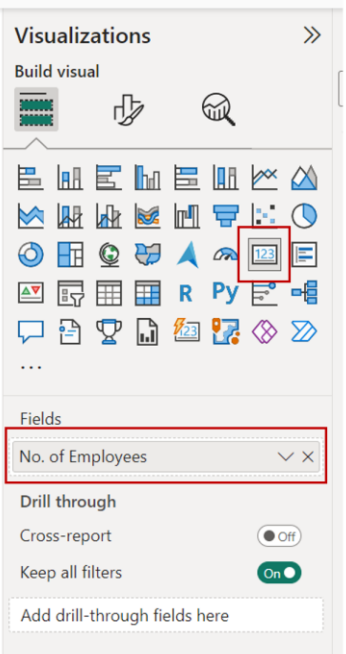
7. Create the **Workforce Summary** Textbox and **Data Table: Number of Employees** Textbox and position as shown here:

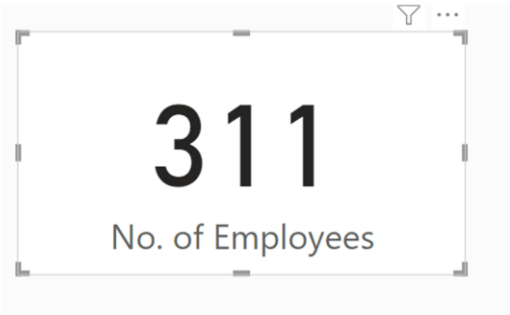


Workforce Summary
Font: Din Light
Font Size: 28
Alignment: Left Align

Data Table: Number of Employees
Font: Din Light
Font Size: 10
Alignment: Left Align

8. Add the Card Visual, using **No. of Employees** as the field for the visual.



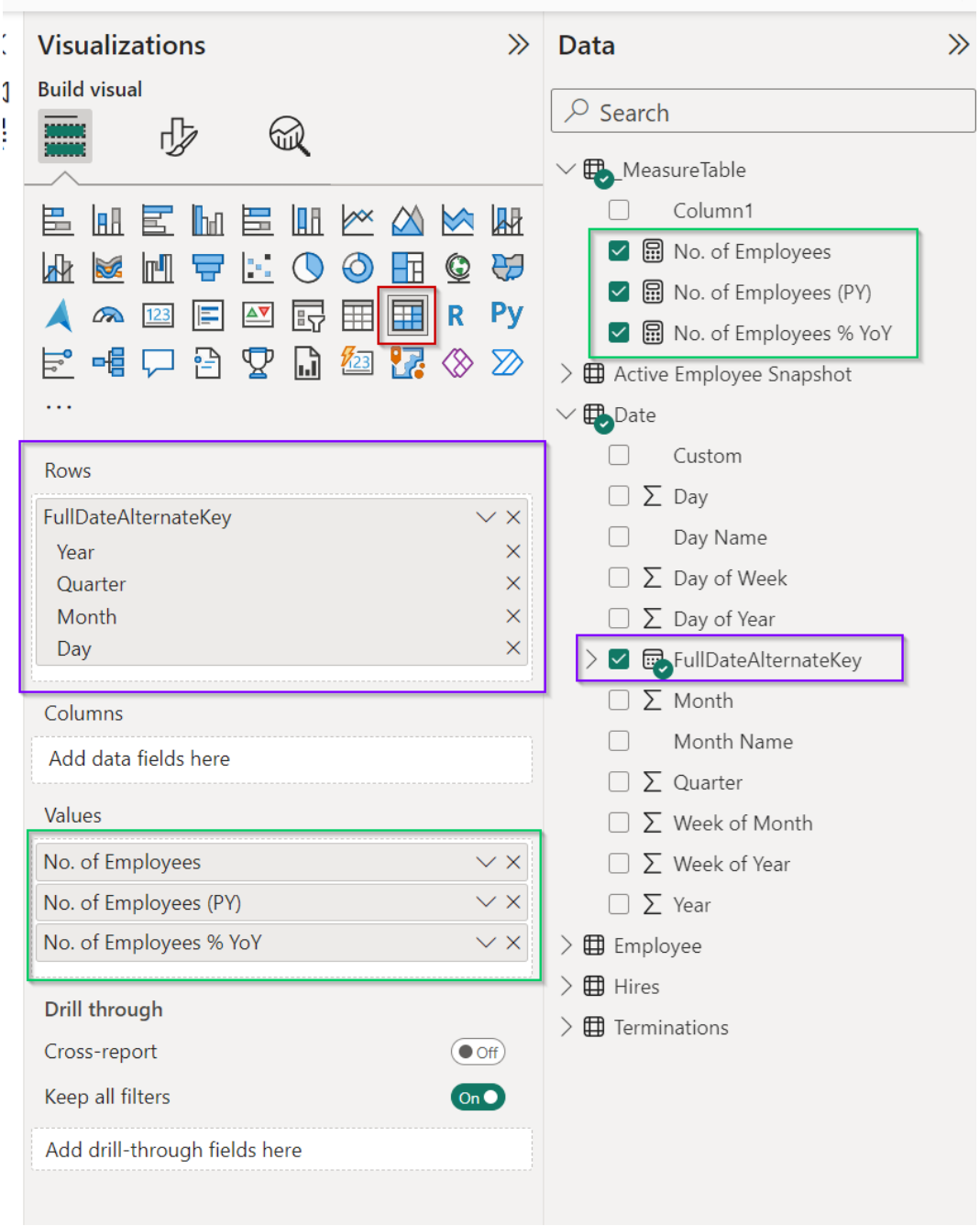


9. Add the Matrix Visual to the Canvas.

Drag or Select the fields into following section:
Rows: FullDateAlternativeKey

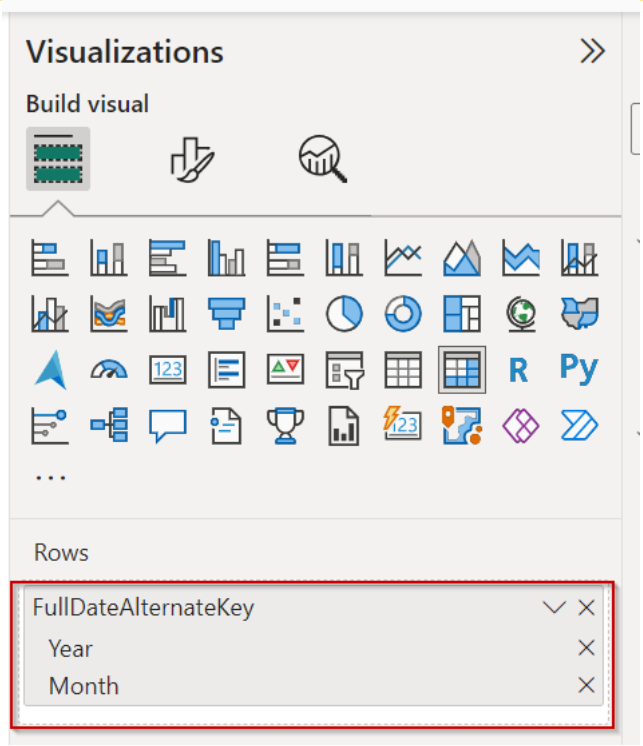
Values:

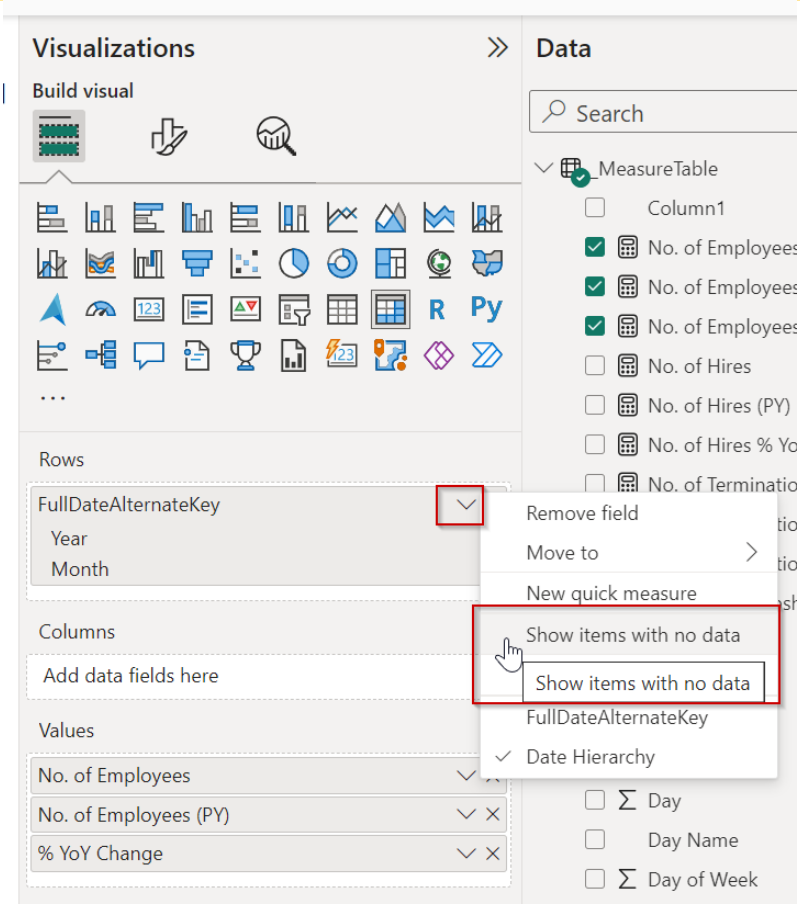
- No. of Employees
- No. of Employees (PY)
- No. of Employees % YoY



10. Remove the **Quarter** and **Day** fields from the Rows section of the Matrix Field by clicking the **X** icon next to each field.

    We won't require them for our visual and simplifies the Matrix field.

	
11.	Click on the dropdown icon on the **FullDateAlternativeKey** field. In the menu that appears, select **Show items with no data**. This will ensure that all months are shown in our matrix, even if there is no data for them. This will be more relevant for the Hires & Termination section of the report.



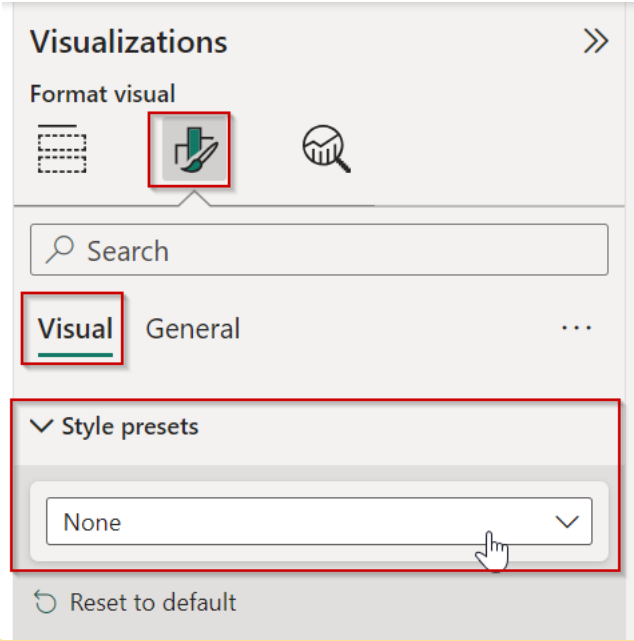
12. Resize the Matrix, and the column width to reassemble the following. You can resize the columns by dragging the borders of the columns.

Data Table: Number of Employees

Year	No. of Employees	No. of Employees (PY)	No. of Employees % YoY
2011	1		
2012	3	1	200.00%
2013	6	3	100.00%
2014	13	6	116.67%
2015	22	13	69.23%
2016	104	22	372.73%
2017	146	104	40.38%
2018	182	146	24.66%
2019	229	182	25.82%
2020	252	229	10.04%
2021	243	252	-3.57%
2022	228	243	-6.17%
2023	221	228	-3.07%
Total	**311**	**310**	**0.32%**

13. Within the Visualisation Pane for the Matrix, navigate to the following area:
Format Visual - > General - > Style Presets

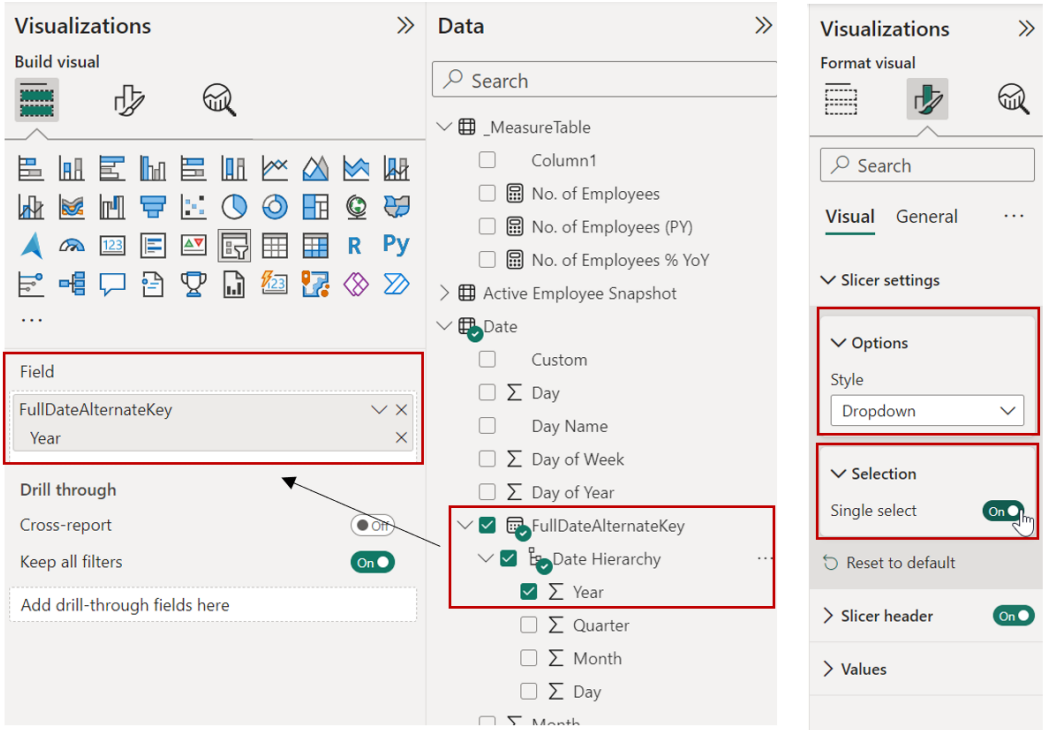
Change the **Style Presets** = None



14. Add a Slicer for the **Year** with the following settings.

Fields: FullDateAlternativeKey - > Year

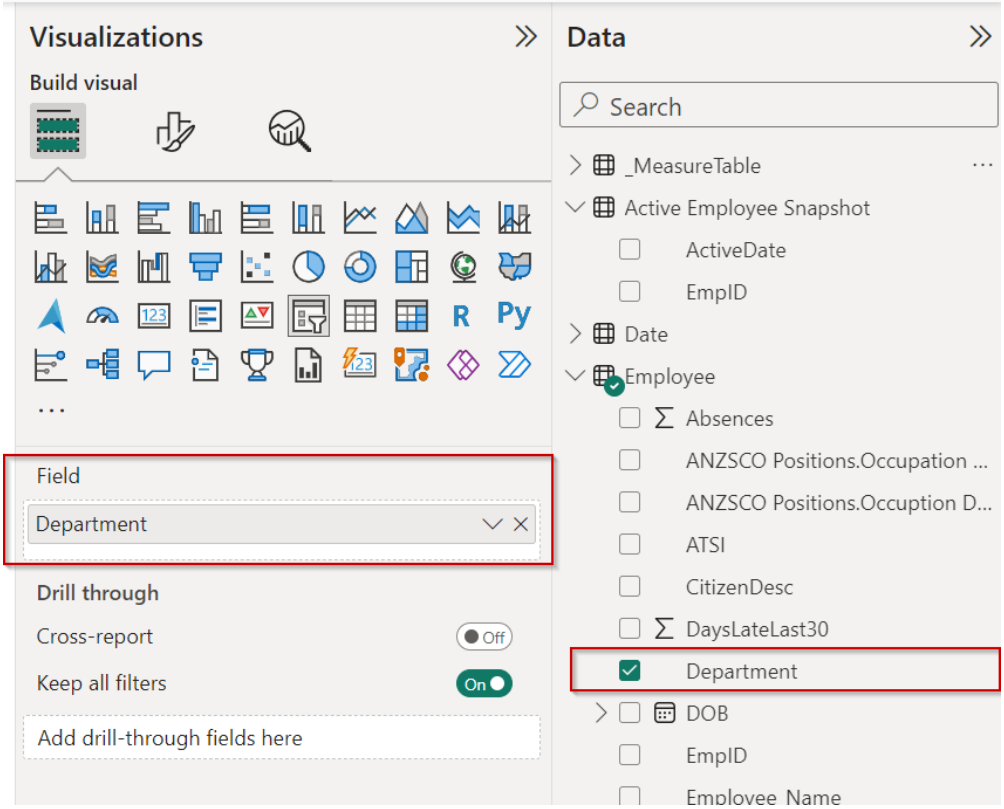
Slicer Settings
Style: Dropdown
Single Select: ON



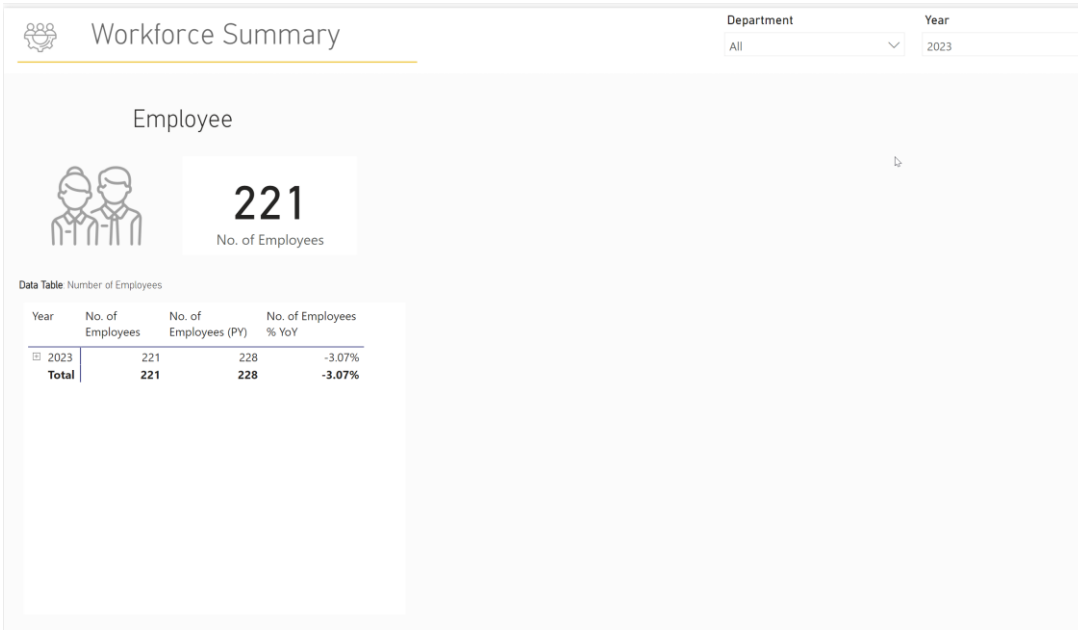
15. Add a Slicer for **Department** with the following settings:

Fields: Department

Slicer Setting:
Style: Dropdown



16. Resize and Position the [Department] and [Year] Slicers to the top right-hand corner as shown here:



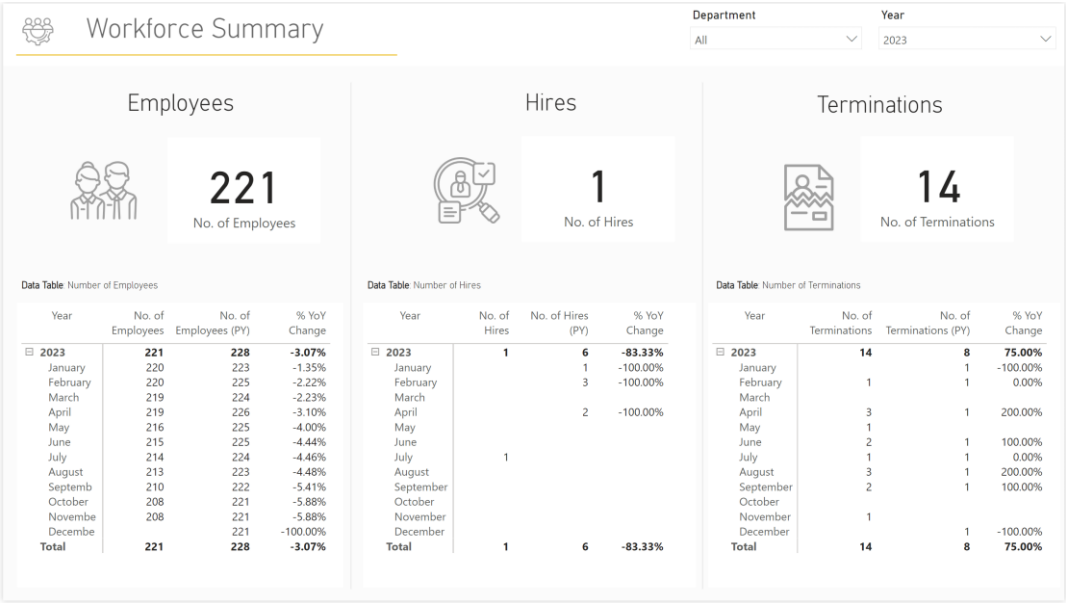
17. Go back to the Matrix and Expand out the rows to reveal the Monthly employee data.

Data Table: Number of Employees

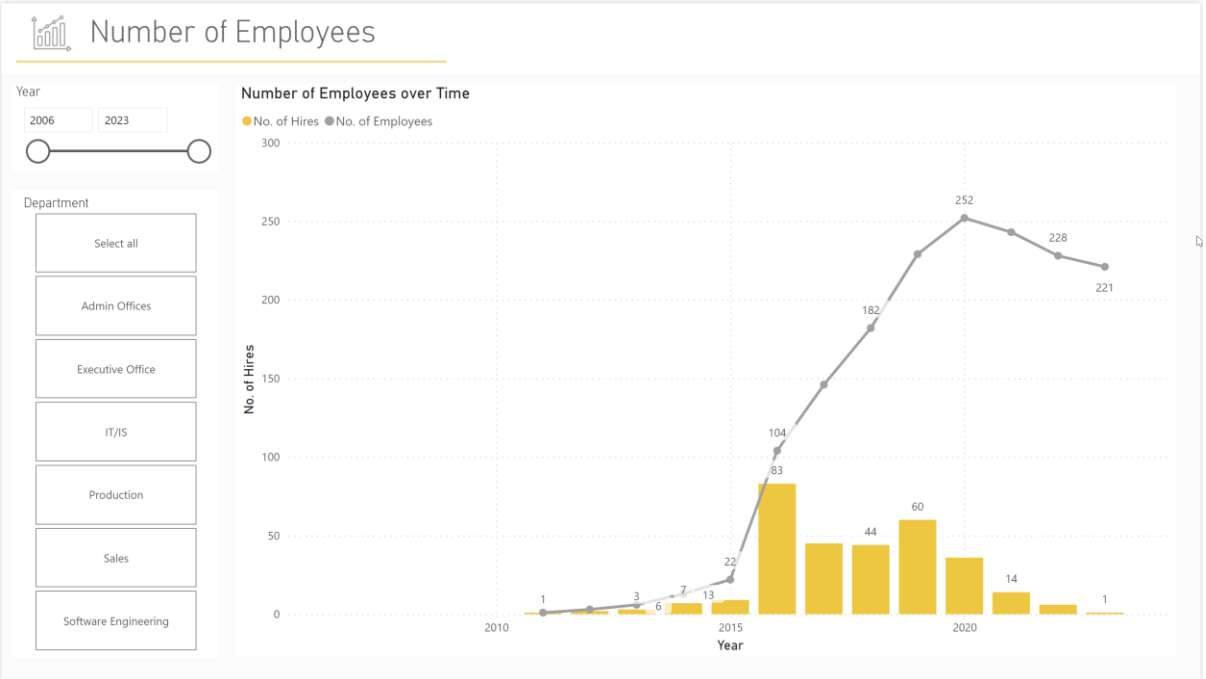
Year	No. of Employees	No. of Employees (PY)	% YoY Change
2023	**221**	**228**	**-3.07%**
January	220	223	-1.35%
February	220	225	-2.22%
March	219	224	-2.23%
April	219	226	-3.10%
May	216	225	-4.00%
June	215	225	-4.44%
July	214	224	-4.46%
August	213	223	-4.48%
Septemb	210	222	-5.41%
October	208	221	-5.88%
Novembe	208	221	-5.88%
Decembe		221	-100.00%
Total	**221**	**228**	**-3.07%**

18. Try and complete the remaining sections of the report.

- Hires
- Terminations



Bonus Exercise: Workforce Trend

Using the Data Models and Measures created in Exercise 3, attempt to re-create the following visualisation.

Note: The main graph used here is a **Line and Stacked Column Chart** visual.