

Power BI Case Study: Sales Analysis and Performance Dashboard

1. Objective:

This case study requires you to create a Sales Performance Dashboard in Power BI to analyze and visualize sales data. The focus will be on key metrics like Total Sales, Profit, Sales Growth, Sales by Region, Sales by Product Category, Sales Trends, and Customer Segmentation. You will need to build measures, calculated columns, and use DAX functions to calculate and visualize this data. The goal is to help businesses gain insights into sales trends, product performance, and customer behavior.

Creating Relationships in Power BI

Before you start creating measures and visualizations, it's essential to set up relationships between tables so Power BI can link them together appropriately.

Here are the steps to create relationships in Power BI for the Sales Analysis case

study: 1. Load the Data into Power BI

1. Open Power BI Desktop.
2. Click on Get Data.
3. Choose the data source (Excel, CSV, SQL, etc.) and select the relevant tables (e.g., Sales, Products, and Dates).
4. Click Load to import the data into Power BI.

2. Open the Model View

1. In Power BI Desktop, click on the Model icon in the left sidebar. It looks like a diagram and is where you can view and manage relationships between your tables.

3. Inspect Tables

You should see the Sales, Products, and Dates tables in the Model view.

Review the fields available in each table:

Sales: Contains ProductID, Date, QuantitySold, SalesAmount, Discount, Profit, SalesChannel, SalesRegion, and CustomerID.

Products: Contains ProductID, ProductName, ProductCategory, Price, CostPrice, and StockQuantity.

Dates: Contains Date, Month, Year, Quarter, MonthName, and

Weekday. **4. Create Relationships**

Now, let's create the necessary relationships between the tables:

4.1 Relationship between Sales and Products

1. Click on the Sales table and locate the ProductID field.
2. Click and drag the ProductID field from the Sales table onto the ProductID field in the Products table.
3. The relationship dialog box will appear. Confirm the following:

Relationship Type: One-to-many (1:*).

The Single Direction relationship will be automatically selected (from Products to Sales).

4. Click OK.

4.2 Relationship between Sales and Dates

1. Click on the Sales table and locate the Date field.
2. Click and drag the Date field from the Sales table onto the Date field in the Dates table.
3. The relationship dialog box will appear. Confirm the following:

Relationship Type: One-to-many (1:*).

Again, the Single Direction relationship will be selected (from Dates to Sales).

4. Click OK.

4.3 Relationship between Products and Dates (Optional)

If you want to analyze sales trends by product category over time, you can also create a relationship between the Products table and the Dates table. However, this is optional and not necessary for basic sales analysis.

1. Click on the Products table and locate the Date field (if applicable, depending on your data structure).
2. Click and drag the Date field from the Products table onto the Date field in the Dates table.

3. Confirm the relationship as One-to-many (1:*) from Dates to Products.

5. Verify Relationships

Once the relationships are created, you should verify them:

1. View the Relationship Diagram in the Model view.
2. Ensure the Sales table has a relationship with both the Products table (via ProductID) and the Dates table (via Date).
3. If there are any issues, such as circular dependencies or ambiguous relationships, Power BI will highlight them with a warning symbol.

6. Set Relationship Properties (Optional)

Cardinality: Most of the time, you will use a one-to-many relationship. However, if you have a many-to-many relationship (for example, if a product can appear in multiple categories), you can configure that accordingly.

Cross Filter Direction: You can choose the filter direction. Single means filters flow only from one table to another (recommended for most cases), while Both allows filtering in both directions. The both direction can be helpful in some complex scenarios but may lead to ambiguity, so use it carefully.

7. Testing Relationships

After creating the relationships:

1. Create a simple table visualization in Power BI:

Drag the ProductName field from the Products table into the rows.

Drag the Total Sales measure from the Sales table into the values.

2. Check if the data appears correctly, ensuring the relationships between the tables work as expected.

8. Save the Data Model

Once the relationships are set up and verified:

Save your Power BI report by clicking File > Save or Save As.

9. Additional Tips

Manage Relationships: If you ever need to modify or view the relationships, go to Model > Manage Relationships. You can add, edit, or delete relationships here.

Use Relationship View: In the Model View, Power BI will display all your relationships. If necessary, you can adjust or delete any existing relationships.

Ensure Data Integrity: Make sure that the keys used in the relationships (such as ProductID, Date) do not contain duplicates in the primary (one) side of the relationship (i.e., in the Products and Dates tables).

By following these steps, you'll be able to set up relationships in Power BI, ensuring your data model works efficiently and allows you to create meaningful visualizations and reports. The relationships connect your data and enable you to perform cross-table analysis.

3. Data Modeling and Calculated Columns:

3.1 Sales After Discount (Calculated Column):

To calculate the final sales amount after applying the discount.

Sales after Discount = Sales[SalesAmount] * (1 - Sales[Discount])

3.2 Profit Margin (Calculated Column):

To calculate the profit margin as a percentage.

Profit Margin = Sales[Profit] / Sales[SalesAmount]

3.3 Sales per Product (Calculated Column):

To calculate sales per product by dividing the sales amount by the quantity sold.

Sales per Product = Sales[SalesAmount] / Sales[QuantitySold]

4. Measures:

4.1 Total Sales (Measure):

To calculate total sales after discount.

Total Sales = SUM(Sales[Sales after Discount])

4.2 Total Profit (Measure):

To calculate total profit.

Total Profit = SUM(Sales[Profit])

4.3 Average Sales per Customer (Measure):

To calculate average sales per customer.

Average Sales per Customer = DIVIDE([Total Sales],
DISTINCTCOUNT(Sales[CustomerID]))

4.4 Average Profit Margin (Measure):

To calculate average profit margin across all sales.

Average Profit Margin = AVERAGE(Sales[Profit Margin])

4.5 Sales by Channel (Measure):

To calculate sales by each channel (Online/In-Store).

Sales by Channel =
CALCULATE([Total Sales], Sales[SalesChannel] = "Online")

4.6 Sales by Product Category (Measure):

To calculate sales by product category (e.g., Electronics, Furniture).

Sales by Category =
CALCULATE([Total Sales], Products[ProductCategory] =

"Electronics") **4.7 Count of Unique Products Sold (Measure):**

To calculate the number of unique products sold.

Distinct Products Sold = DISTINCTCOUNT(Sales[ProductID])

5. Visualizations:

5.1 Total Sales by Region (Bar Chart):

Use the SalesRegion field from the Sales table and Total Sales measure to create a Bar Chart showing total sales by region.

5.2 Sales by Channel (Pie Chart):

Use SalesChannel as the Legend and Total Sales as the Values to create a Pie Chart representing the distribution of sales across different sales channels (Online, In-Store).

5.3 Average Sales per Customer (Card Visualization):

Display the Average Sales per Customer using a Card visualization.

5.4 Sales Trend (Line Chart):

Use the Date field from the Dates table and Total Sales measure to create a Line Chart showing total sales over time.

5.5 Sales by Product Category (Stacked Bar Chart):

Use ProductCategory for the Axis and Sales per Product for the Values to display Sales by Product Category using a Stacked Bar Chart.

5.6 Profit vs Sales (Scatter Chart):

Create a Scatter Chart to plot Total Sales on the X-axis and Total Profit on the Y-axis. This helps you visualize the relationship between profit and sales.

5.7 Total Sales by Month (Column Chart):

Use the MonthName field from the Dates table and Total Sales measure to create a Column Chart showing total sales by month.

5.8 Product Sales Performance (Tree Map):

Create a Tree Map to visualize the sales distribution by ProductName. This allows you to see the top-selling products.

5.10 Sales per Customer (Line Chart):

Create a Line Chart that shows Average Sales per Customer over time (by month or

week). **6. Additional Visualizations:**

6.1 Sales by Sales Channel and Region (Stacked Column Chart):

To compare sales across different Sales Channels and Regions, use a Stacked Column Chart.

6.2 Sales by Product (Line Chart):

Visualize product sales performance over time using a Line Chart.

6.3 Customer Segmentation (Pie Chart):

Use a Pie Chart to represent customer distribution across different categories.

6.4 Top Products (Table):

Create a Table Visualization that shows the top-selling products, filtered by Total Sales using the RankX function.