# **Project Overview**

**Project Name:**  Superstore sales analysis

**Objective:** Analyse the comparisons of every year’s sales data and create visuals that gives an overview of sales analysis

**Data Sources Used**: the source is superstore csv file

**Data Import Method:** order imported into power BI form import mode -> selecting CSV -> uploading from the location where csv file is saved

**1. Observation Data (Raw Data) :**

* **Description of Raw Data:** 
  + **The data has 21 columns and 9995 rows(including header).**
  + **Important indicative columns that can provide insights are “Order ID”, “Order Date”, ”Region”, “Sales”, “Profit”. “Order ID” is repeated sometimes as a single order may contains multiple products.**
* **Initial Data Inspection:**
  + **Used Power BI’s Data view to check for missing values, duplicates, or inconsistent data formats.**
  + **Observed that some fields contained null values, and columns like “Order Date”, “Ship Date” had incorrect/inconsistent date formats thought the data.**

**2. Data Reformatting**

* **Date Formats:**
  + **columns like “Order Date”, “Ship Date” were not in a consistent format. Dates were in 11-05-2016 (DD-MM-YYYY) & 11/24/2016 (MM/DD/YYYY) format, need to unify this in “DD-MM-YYYY” format.**
  + **Used Power Query transformations to ensure all date fields were correctly parsed as Date/Time data types.**
  + **Steps followed : In Power BI, Home > Transform Data, clicked on the “Order Date” column. Then Transform > Data Type > Text. Then added a Custom Column “CleanOrderDate” and used power query mentioned below and applied , after that Transform > Data Type > Date. In Modeling tab, Under Format selected: “DD-MM-YYYY” format.**

**let**

**raw = [OrderDate],**

**// Standardize separator to "/"**

**standardized = Text.Replace(raw, "-", "/"),**

**// Split into parts**

**parts = Text.Split(standardized, "/"),**

**// Try to detect if format is MM/DD/YYYY or DD/MM/YYYY**

**month = Number.FromText(parts{0}),**

**day = Number.FromText(parts{1}),**

**year = Number.FromText(parts{2}),**

**// Heuristic: if month > 12, it's probably DD/MM/YYYY → swap**

**correctedDate = if month > 12 then #date(year, day, month) else #date(year, month, day)**

**in**

**correctedDate**

* + **Example:**

**DateColumn = Table.TransformColumns(YourTable, {{"DateColumn", each Date.FromText(\_), type date}})**

* **Text to Date Conversion:**
  + **Some text fields represented dates in non-standard formats. These were converted using Date.FromText function.**

**3. Data Cleaning**

* **Removing Duplicates:**
  + **Used Power Query to remove duplicates based on [list columns, e.g., "Sales ID," "Date," etc.].**
  + **Steps:**
    1. **Selected the relevant columns.**
    2. **Clicked "Remove Duplicates" in Power Query Editor.**
* **Handling Missing Values:**
  + **For missing numeric values, I used the Fill Down or Fill Up function in Power Query where applicable.**
  + **For missing text values, I used "Replace Values" to replace nulls with "Unknown."**
  + **Example:**

**#"Replaced Nulls" = Table.ReplaceValue(#"Previous Step", null, "Unknown", Replacer.ReplaceValue, {"Column Name"})**

* **Outliers/Errors:**
  + **Detected outliers using basic statistics (mean, median) and fixed them using DAX or Power Query transformations.**

**4. Data Transformation**

* **Creating New Columns:**
  + **Created new calculated columns using Power Query or DAX formulas.**
  + **Example: Created a "Year" column from a "Date" field.**

**dax**

**CopyEdit**

**Year = YEAR('Sales'[Date])**

* **Aggregating Data:**
  + **Aggregated sales by region and month to get the total sales per month per region.**
    - **Used DAX formula:**

**dax**

**CopyEdit**

**SalesPerRegion = SUM('Sales'[SalesAmount])**

* **Relationships:**
  + **Defined relationships between tables [Table1 to Table2] using common fields like "Region" or "ProductID."**
  + **Ensured the data model was set up with proper relationships (one-to-many, many-to-one, etc.).**

**5. Data Transformation , DAX Formulas Used and Purposes as per requirements**

* **Key Measures and Calculated Columns:**
  + **Total Sales that should not filter by selection slicers:**

**DAX -> Total\_Sales = CALCULATE(SUM(‘Superstore’[Sales]),ALL(‘Superstore’)) OR**

**DAX -> Total\_Sales = SUMX(ALL(‘Superstore’), ‘Superstore’[Sales])**

* + **Current Year Sales:**

**DAX -> CY\_Sales = SUM(‘Superstore’[Sales])**

* + **Previous Year Sales:**

**DAX -> PY\_Sales = CALCULATE([CY],SAMEPERIODLASTYEAR(‘Superstore’[“CleanOrderDate”].[Date])**

* + **YOY%(Year over Year) on Sales:**

**DAX -> YOY\_Sales = DIVIDE((([CY\_Sales]-[ PY\_Sales])/[ PY\_Sales]) \* 100)**

* + **Total no. of customers in each year on slicer selections :**

**DAX -> Total\_Customers = DISTINCTCOUNT('Superstore'[Customer Name])**

* + **Total no. of customers in each year on slicer selections :**

**DAX -> Total\_Customers = DISTINCTCOUNT('Superstore'[Order ID])**

* + **Calculate Top 5 sales by customers :**

**Top\_5\_Sales\_by\_Customers =**

**CALCULATE([CY\_Sales],TOPN(5,VALUES('Superstore'[Customer Name]),[Total Sales],DESC))**

**Note - Visual/Table with Top 5 Customers**

**If you're building a table visual in Power BI, and want to show only top 5 customers by sales:**

* **Add Customer Name and [Total Sales] to the table.**
* **Then use the Visual Filter pane:**
  + **Filter on [Total Sales]**
  + **Choose Top N**
  + **Enter 5**
  + **By value: [Total Sales]**

**dax**

**CopyEdit**

**YTD Sales = TOTALYTD(SUM('Sales'[SalesAmount]), 'Sales'[Date])**

* + **Average Sales:**

**dax**

**CopyEdit**

**AverageSales = AVERAGE('Sales'[SalesAmount])**

* **Purpose of DAX:**
  + **To calculate key business metrics like Total Sales, YTD Sales, and Average Sales.**
  + **To handle time intelligence, like computing monthly or quarterly sales growth.**

**6. Visualizations**

* **Charts:**
  + **Bar/Column Charts: Used to display total sales by region.**
  + **Line Chart: Used to visualize sales trends over time (monthly, quarterly).**
  + **Pie Chart: Used to show sales distribution across different product categories.**
* **Treemaps:**
  + **Used for visualizing hierarchical data, such as sales by region, then by product category.**
  + **Purpose: To get a quick view of how each region and product category contributes to total sales.**
* **Matrices:**
  + **A matrix visualization was used to display sales across regions and time periods (e.g., months or years).**
  + **Purpose: To give a detailed and drillable view of the data in a tabular format.**
* **Slicers and Filters:**
  + **Used slicers for filtering by region, date, or product category.**
  + **Interactive slicers allowed users to filter the report to focus on specific areas of interest.**

**7. Conclusion and Insights**

* **Key Insights Derived:**
  + **Identified that sales in Region A had a significant increase in Q2.**
  + **Product Category X has the highest sales but also the most returns.**
* **Next Steps:**
  + **Add more historical data for deeper analysis.**
  + **Consider integrating external market data to enhance insights.**