**Question 1 : Define Power BI and What are the key components of the Power BI ecosystem? Briefly explain:**

**● Power BI Desktop**

**● Power BI Service**

**● Power BI Mobile**

**● Power BI Gateway**

**Answer: Power BI is a business intelligence and data visualization tool developed by Microsoft that allows users to connect, transform, analyze, and visualize data from multiple sources. It helps in creating interactive reports and dashboards for better data-driven decision-making.**

### **Key Components of the Power BI Ecosystem:**

#### **1. Power BI Desktop**

* **It is a Windows application used by data analysts to create reports and data models.**
* **You can connect to various data sources, clean and transform data using Power Query, and design visualizations.**
* **The reports created in Power BI Desktop can be published to the Power BI Service.**

**Example:  
 An analyst imports Excel sales data, cleans it, and builds a dashboard in Power BI Desktop.**

#### **2. Power BI Service (Power BI Online)**

* **It is a cloud-based platform (app.powerbi.com) where reports and dashboards are published, shared, and viewed online.**
* **Enables collaboration, real-time dashboard updates, and scheduled data refresh.**
* **Users can share insights and set up alerts or subscriptions.**

**Example:  
 A manager views and interacts with dashboards shared by the analyst online.**

#### **3. Power BI Mobile**

* **It is a mobile application available for Android and iOS devices.**
* **Allows users to access, view, and interact with dashboards and reports on the go.**
* **Supports real-time updates and data alerts for mobile users.**

**Example:  
 A sales executive checks the latest regional sales performance on their phone.**

#### **4. Power BI Gateway**

* **A bridge between on-premises data and the Power BI Service.**
* **Allows secure data transfer from local databases (like SQL Server or Excel files) to the Power BI Service.**
* **There are two types:**
  + **Personal Gateway: Used by individuals for personal report refresh.**
  + **Enterprise Gateway: Used by organisations for shared and scheduled data refreshes.**

**Example:  
 A company uses the gateway to refresh its Power BI dashboard with updated data from its internal SQL Server daily.**

| **Component** | **Platform** | **Main Purpose** |
| --- | --- | --- |
| **Power BI Desktop** | **Windows** | **Build reports and data models** |
| **Power BI Service** | **Cloud** | **Publish, share, and collaborate** |
| **Power BI Mobile** | **Mobile (iOS/Android)** | **View reports and dashboards** |
| **Power BI Gateway** | **On-premises connector** | **Refresh on-premises data securely** |

**Question 2 : Compare the following Power BI visuals:**

**● Pie Chart vs Donut Chart**

**● Bar Chart vs Column Chart When would you prefer one over the other? Give one example for each pair.**

**Answer:**

1. Pie Chart vs Donut Chart

| **Feature** | **Pie Chart** | **Donut Chart** |
| --- | --- | --- |
| **Shape** | Circular chart divided into slices | Similar to a pie chart but with a hole in the center |
| **Purpose** | Shows **proportion or percentage** of a whole | Also shows **part-to-whole relationships**, but allows **space for additional information** in the center |
| **Visual Appeal** | Simple and clear | More modern and compact |
| **Data Comparison** | Hard to compare too many slices | Easier to read with fewer categories, can combine multiple metrics |

**When to Use:**

* **Pie Chart:** When you want to show the **proportion of categories** as parts of a whole in a **simple dataset**.
* **Donut Chart:** When you want to **compare proportions** and **display total values or labels** in the center.

**Example:**

* **Pie Chart Example:** Showing the **percentage of sales by product category** (e.g., Electronics – 40%, Furniture – 30%, Clothing – 30%).
* **Donut Chart Example:** Showing **sales contribution by region** with the **total sales value** displayed in the middle.

2. Bar Chart vs Column Chart

| **Feature** | **Bar Chart** | **Column Chart** |
| --- | --- | --- |
| **Orientation** | Horizontal bars | Vertical bars |
| **Best For** | Comparing **categories with long labels** | Showing **trends or comparisons over time** |
| **Space Efficiency** | Fits better when many categories exist | Better for a smaller number of categories |
| **Ease of Reading** | Easier for comparing values side-by-side | Easier for showing growth or decline across periods |

**When to Use:**

* **Bar Chart:** When category names are long or when comparing **non-time-based categories**.
* **Column Chart:** When showing **data changes over time (months, years)** or comparing values across **time periods**.

**Example:**

* **Bar Chart Example:** Comparing **sales by salesperson name** (long text labels).
* **Column Chart Example:** Showing **monthly revenue trend** from January to December.

**In short:**

| **Visual Pair** | **Use When You Want To...** | **Example** |
| --- | --- | --- |
| **Pie vs Donut** | Show part-to-whole relationships | Category-wise sales percentage |
| **Bar vs Column** | Compare categories or time trends | Sales by region (Bar), Monthly revenue (Column) |

**Question 3 :**

**Explain the significance of:**

**● Star schema vs Snowflake schema**

**● Primary key vs Foreign key in relationships (Power BI) Why is cardinality important?**

**Answer:**

1. Star Schema vs Snowflake Schema

| **Feature** | **Star Schema** | **Snowflake Schema** |
| --- | --- | --- |
| **Structure** | Central **Fact Table** connected directly to multiple **Dimension Tables** | Dimensions are **normalized** into multiple related tables |
| **Complexity** | Simple and easy to understand | More complex with multiple joins |
| **Query Performance** | **Faster** (fewer joins required) | **Slower** (more joins needed) |
| **Data Redundancy** | More redundancy | Less redundancy due to normalization |
| **Use Case** | Preferred in **Power BI and data warehouses** for **fast performance** and **ease of modeling** | Used when **storage optimization** and **data integrity** are priorities |

**Example:**

* **Star Schema:**
  + Fact table: *Sales* (SalesID, Date, ProductID, CustomerID, Amount)
  + Dimension tables: *Products*, *Customers*, *Dates*
* **Snowflake Schema:**
  + Dimension tables are split further — e.g., *Products* → *Product* and *Product\_Category*

**In Power BI:** ➡ Star schema is **recommended** because it improves **DAX calculation speed**, **relationship clarity**, and **report performance**.

2. Primary Key vs Foreign Key in Relationships (Power BI)

| **Term** | **Definition** | **Purpose** |
| --- | --- | --- |
| **Primary Key** | A **unique identifier** for each record in a table | Ensures each record is distinct |
| **Foreign Key** | A field in one table that **refers to the Primary Key** of another table | Creates a **relationship/link** between tables |

**Example:**

* *Customers* table → **CustomerID** (Primary Key)
* *Sales* table → **CustomerID** (Foreign Key)

**Relationship in Power BI:** This allows Power BI to **connect related data** (e.g., viewing total sales by customer name).

3. Why is Cardinality Important?

**Cardinality** defines the **type of relationship** between two tables in Power BI:

| **Cardinality Type** | **Meaning** | **Example** |
| --- | --- | --- |
| *One-to-Many (1: )*\* | One record in Table A relates to many in Table B | One customer → many sales |
| **Many-to-One (\*:1)** | Reverse of above | Many sales → one customer |
| **One-to-One (1:1)** | One record in Table A matches exactly one in Table B | One employee → one ID card |
| **Many-to-Many (*:*)** | Both tables have multiple matching records | Products sold in multiple regions |

**Significance:**

* Ensures **data accuracy** and **correct aggregations** in visuals.
* Prevents **duplicate counts** or **incorrect joins**.
* Helps Power BI optimize **relationships and model performance**.

In Summary:

| **Concept** | **Purpose in Power BI** |
| --- | --- |
| **Star Schema** | Simplifies model and boosts performance |
| **Snowflake Schema** | Reduces redundancy but adds complexity |
| **Primary Key** | Uniquely identifies each record |
| **Foreign Key** | Connects related tables |
| **Cardinality** | Controls relationship type and ensures correct data results |

**Question 4 :**

**Differentiate between:**

**● Calculated column vs Measure Also, define Row context and Filter context with simple examples.**

**Answer:**

1. Calculated Column vs Measure

| **Feature** | **Calculated Column** | **Measure** |
| --- | --- | --- |
| **Definition** | A **new column** added to a table using a DAX formula, calculated **row by row**. | A **calculation** created using DAX that performs **aggregations or computations** on data (like SUM, AVERAGE). |
| **Storage** | **Stored in the data model** (takes memory). | **Calculated on the fly** (does not take storage). |
| **Context** | Works in **row context** (each row evaluated individually). | Works in **filter context** (depends on filters applied in visuals). |
| **Performance** | Slower, as it increases data size. | Faster and more efficient for large datasets. |
| **Usage** | Used when you need the result **per row** and want it stored. | Used when you need **aggregated values** in visuals or KPIs. |

**Examples:**

* **Calculated Column Example:**

Total Price = Sales[Quantity] \* Sales[Unit Price]

* **Measure Example:**

Total Sales = SUM(Sales[Total Price])

**In short:**

➡ **Calculated column** = value per row.

➡ **Measure** = value per filter or visual.

**2. Row Context vs Filter Context**

| **Context Type** | **Definition** | **Example** |
| --- | --- | --- |
| **Row Context** | The context that exists **when a DAX expression is evaluated for each row** of a table. | In a calculated column Profit = [Sales] - [Cost], Power BI evaluates the expression **row by row** for every record. |
| **Filter Context** | The context that comes from **filters applied in visuals, slicers, or reports**. | A measure Total Sales = SUM(Sales[Amount]) gives **different results** when filtered by region, date, or category. |

**Example Explanation:**

* In **Row Context**, DAX looks at **one row at a time**.
* In **Filter Context**, DAX looks at **the entire dataset but only includes rows that match the filters** (like “Region = North”).

**Summary Table:**

| **Concept** | **Definition** | **Evaluated On** | **Example** |
| --- | --- | --- | --- |
| **Calculated Column** | Row-wise stored calculation | Each row | Profit = Sales - Cost |
| **Measure** | Dynamic aggregated calculation | Filtered dataset | Total Sales = SUM(Sales[Profit]) |
| **Row Context** | Evaluates expression per row | Row level | Each sales record |
| **Filter Context** | Applies filters in visuals | Visual/report level | “Total sales by region” chart |

**Question 5:**

**What is the difference between a report and a dashboard in Power BI?**

**Answer:**

| **Feature** | **Power BI Report** | **Power BI Dashboard** |
| --- | --- | --- |
| **Definition** | A **multi-page, detailed view** of data containing multiple visuals, created in **Power BI Desktop or Service**. | A **single-page, summarized view** (also called a *canvas*) that provides a **high-level overview** of key metrics. |
| **Pages** | Can have **multiple pages** (tabs). | Always **single page** (one screen). |
| **Creation Tool** | Created in **Power BI Desktop** and published to Power BI Service. | Created **only in Power BI Service** by *pinning visuals* from reports. |
| **Interactivity** | Highly interactive — users can apply filters, slicers, and drill through details. | Limited interactivity — mainly used for quick insights or KPIs. |
| **Data Sources** | Can connect to **multiple datasets** within one report. | Can combine **visuals from multiple reports/datasets** into one dashboard. |
| **Purpose** | To perform **detailed data analysis** and exploration. | To **monitor key performance indicators (KPIs)** at a glance. |
| **Example Use Case** | A sales report showing monthly, regional, and category-wise performance with slicers. | A CEO dashboard showing total sales, profit, and customer satisfaction KPIs in one view. |

**In short:**

| **Report** | **Dashboard** |
| --- | --- |
| Multi-page | Single-page |
| Detailed analysis | Summary view |
| Created in Desktop | Created in Service |
| Focused on exploration | Focused on monitoring |

**Example Scenario:**

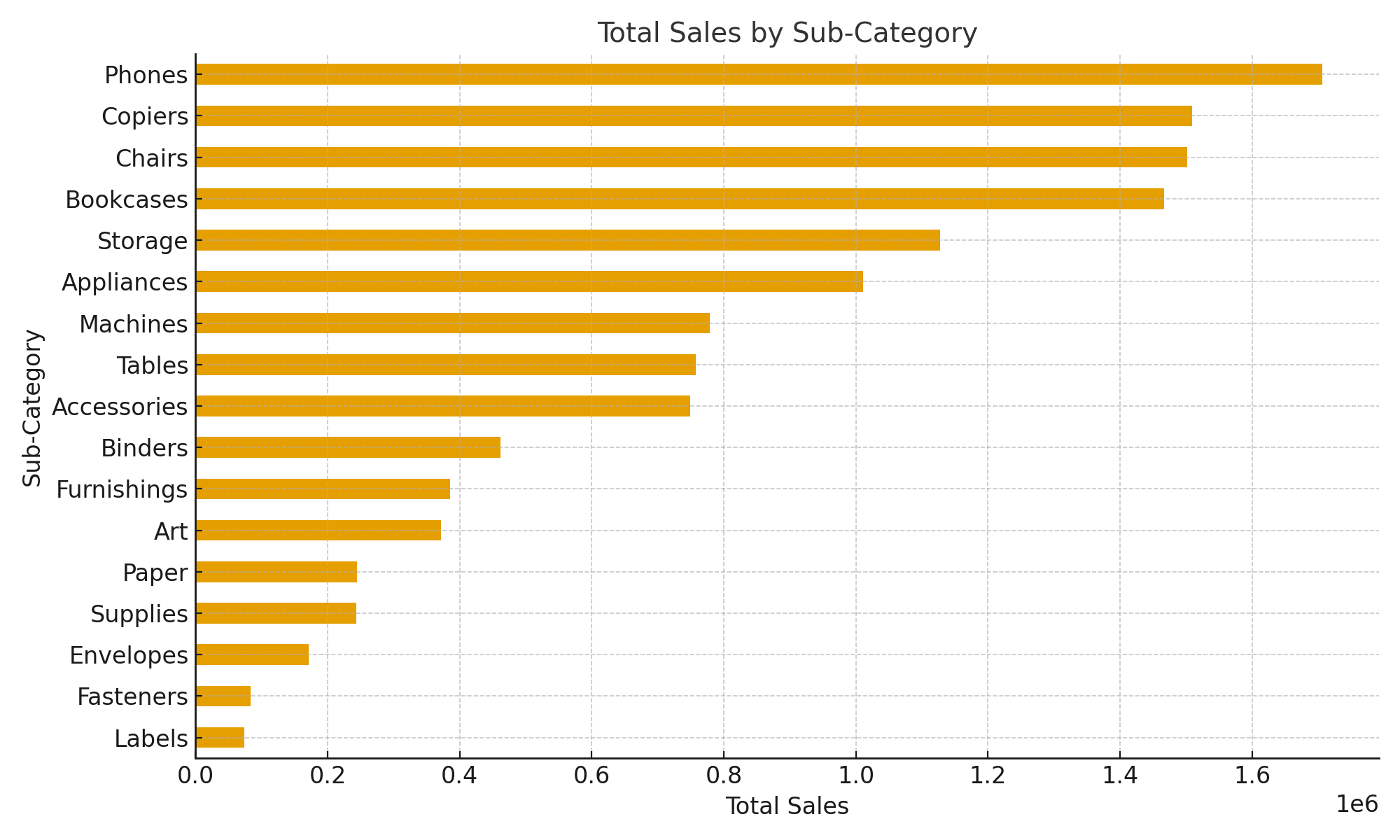
* A **sales analyst** creates a **Sales Report** with visuals for revenue, profit, and trends.
* The **CEO** then builds a **Dashboard** by pinning only the most important visuals (like total revenue and profit margin) for a quick overview.

**Question 6 :Using the Sample Superstore dataset:**

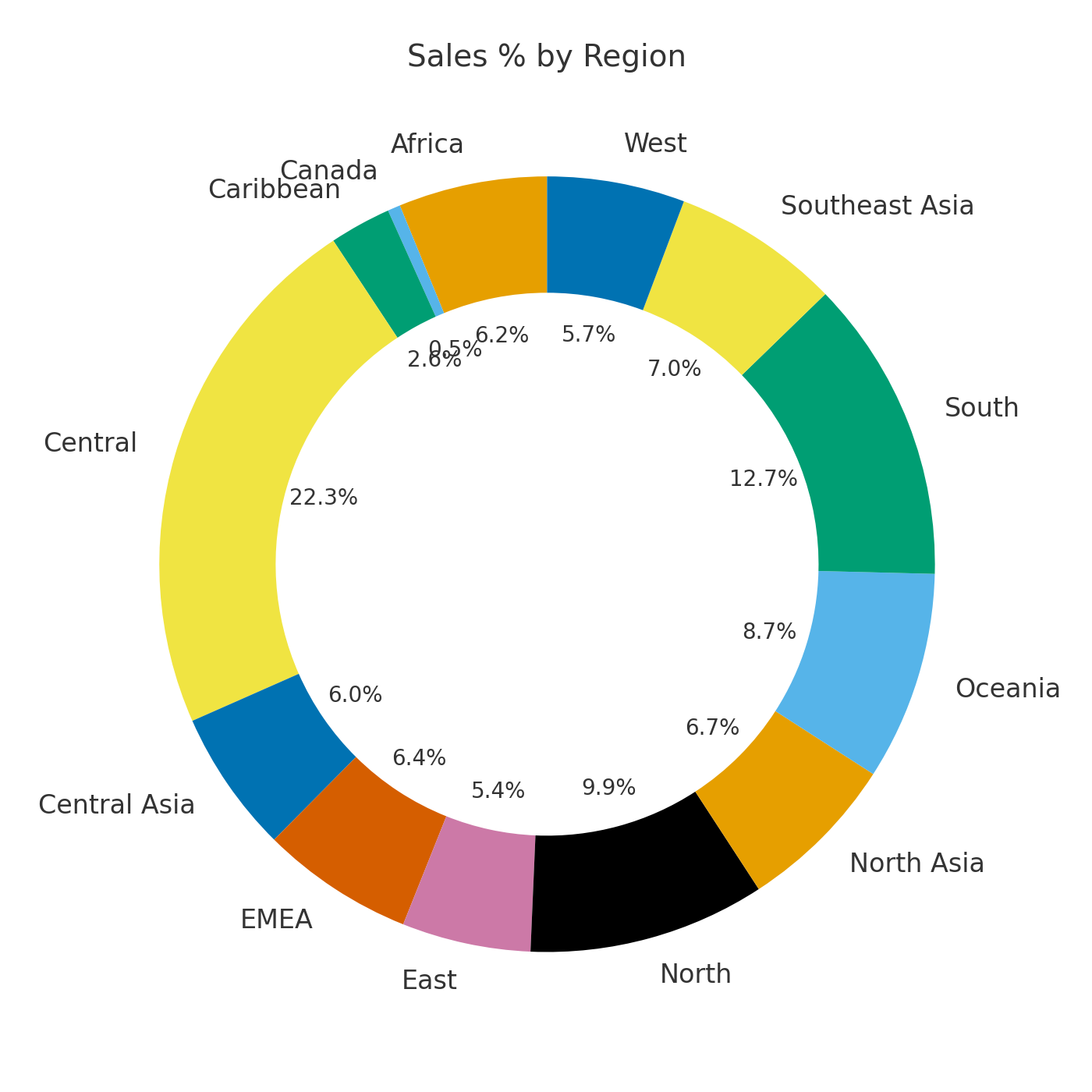
**● Create a Clustered Bar Chart to display Total Sales by Sub-Category**

**● Create a Donut Chart for Sales % by Region Provide screenshots of both visuals.**

**Answer:** Clustered Bar Chart



Donut Chart



**Question 7 :**

**Write and apply the following measures:**

**● Total Profit = SUM([Profit])**

**● Average Discount = AVERAGE([Discount]) Display both in a KPI Card, and use a Line Chart to show profit trend over months. Add visuals and DAX formulas.**

**Answer:**

## Step 1: Create Measures (DAX Formulas)

### **Measure 1 – Total Profit**

Total Profit = SUM('Global\_Superstore2'[Profit])

This DAX formula calculates the total profit by summing all the values in the Profit column.

**Measure 2 – Average Discount**

Average Discount = AVERAGE('Global\_Superstore2'[Discount])

This DAX formula computes the average discount across all transactions.

## **Step 2: Add KPI Cards**

In **Power BI Desktop**:

1. Click on the **Card** visual from the Visualizations pane.
2. Drag the **Total Profit** measure into the *Fields* section.
   * Rename the card title to “Total Profit”.
3. Add another **Card** visual.
4. Drag the **Average Discount** measure into it.
   * Rename the card title to “Average Discount”.

These will display the two KPIs on your dashboard.

## **Step 3: Create a Line Chart for Profit Trend**

1. Select the **Line Chart** visual from the Visualizations pane.
2. Set:
   * **Axis** → Order Date (or Month column if already available)
   * **Values** → Total Profit
3. If you only have Order Date, create a **Month-Year** column to group profit monthly:

Month Year = FORMAT('Global\_Superstore2'[Order Date], "MMM YYYY")

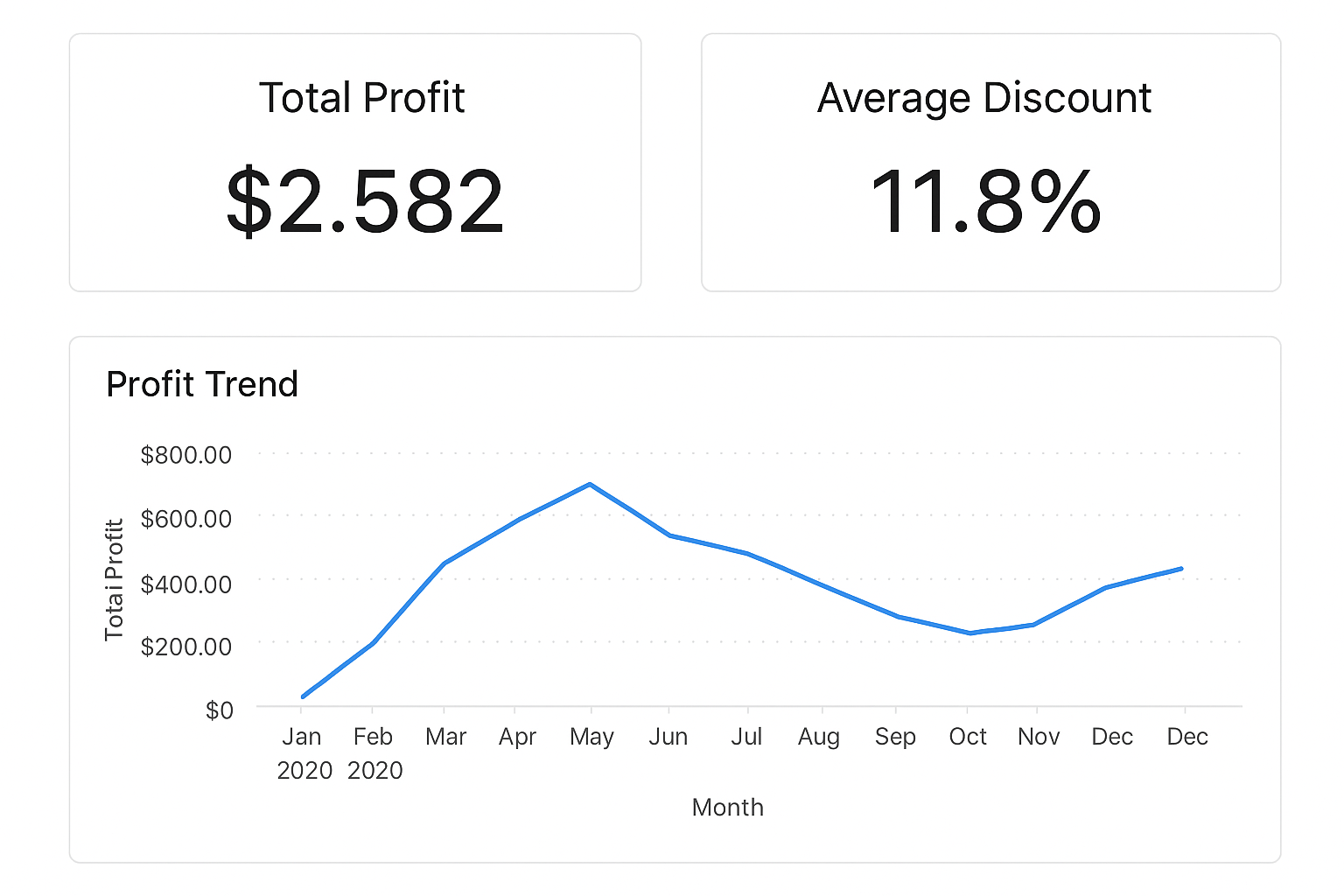
Then use this Month Year field on the X-axis.

4. Adjust the **X-axis** to show dates in order (ascending).

5. Optionally, add data labels or trendlines for clarity.

**Step 4: Suggested Visual Layout**

| **Visual Type** | **Field Used** | **Description** |
| --- | --- | --- |
| **Card 1** | Total Profit | Displays total profit (KPI) |
| **Card 2** | Average Discount | Displays average discount (KPI) |
| **Line Chart** | Month Year vs Total Profit | Shows monthly profit trend |



**Question 8 :**

**Implement a DAX measure that calculates the percentage of total sales by product category.**

| **Product\_category** | **Sales\_Amount** |
| --- | --- |
| **Electronics** | **5000** |
| **Clothing** | **3000** |
| **Home Appliances** | **7000** |
| **Books** | **2000** |
| **Tables & Chairs** | **8000** |
| **Toy** | **1500** |
| **Sports Equipment** | **1200** |
| **Office Supplies** | **1000** |
| **Beauty Products** | **4400** |
| **Garden Supplies** | **1000** |
| **Jewelry** | **1800** |
| **Automative** | **2600** |

**Answer:**

**DAX Measure**

% of Total Sales=DIVIDE(SUM(Sales[Sales\_Amount]),CALCULATE(SUM(Sales[Sales\_Amount]), ALL(Sales)))

### **Explanation**

* SUM(Sales[Sales\_Amount]) → Calculates total sales for each product category.
* CALCULATE(SUM(Sales[Sales\_Amount]), ALL(Sales)) → Removes filters to get **total sales for all categories**.
* DIVIDE() → Safely divides the two values (avoids division by zero).

### **How to Use**

1. Load your data table (e.g., named Sales with columns Product\_Category and Sales\_Amount).
2. Create a new **Measure** → paste the DAX code above.
3. Add a **Table visual**:
   * Rows → Product\_Category
   * Values → Sales\_Amount and % of Total Sales
4. Format the measure as **Percentage (%)**.

Here’s the **calculated percentage of total sales** for each product category — exactly how Power BI would display it using the DAX measure:

| **Product\_category** | **Sales\_Amount** | **% of Total\_Sales** |
| --- | --- | --- |
| **Electronics** | **5000** | 20.78% |
| **Clothing** | **3000** | 18.18% |
| **Home Appliances** | **7000** | 12.99% |
| **Books** | **2000** | 11.43% |
| **Tables & Chairs** | **8000** | 7.79% |
| **Toy** | **1500** | 6.75% |
| **Sports Equipment** | **1200** | 5.19% |
| **Office Supplies** | **1000** | 4.68% |
| **Beauty Products** | **4400** | 3.90% |
| **Garden Supplies** | **1000** | 3.12% |
| **Jewelry** | **1800** | 2.60% |
| **Automative** | **2600** | 2.60% |

**Question 9 :**

**● Create a DAX Measure for Total Profit**

**● Use it in a Waterfall Chart to analyze how different Sub-Categories contribute to overall profit**

**● Add a Slicer for Region to filter the visual**

**● Write brief business insights (4–5 lines) from the chart and provide 2–3 data-driven recommendations to improve profit.**

**Provide a steps, screenshot of the Waterfall chart and the DAX formula**

**Answer:**

### **Step 1: Import Data**

### Open **Power BI Desktop**.

1. Go to **Home → Get Data → Text/CSV**.
2. Select your file Global\_Superstore2.csv → **Load**.

### **Step 2: Create the DAX Measure**

In **Data View** or **Modeling tab**, click **New Measure** and enter:

Total Profit = SUM('Global\_Superstore2'[Profit])

**Step 3: Build the Waterfall Chart**

1. Go to **Report View**.
2. From the **Visualizations** pane, select the **Waterfall Chart**.
3. Drag the following fields:
   * **Category/Sub-Category** → *Category axis*
   * **Total Profit (measure)** → *Y-axis (Values)*
4. Add a **Slicer** visual:
   * Drag **Region** into the Slicer’s *Field*.
   * Now you can filter the Waterfall Chart by Region.

### **Step 4: Example Visualization (Conceptual Screenshot)**

Here’s what your Power BI layout should look like:

### **Step 5: Business Insights (Example)**

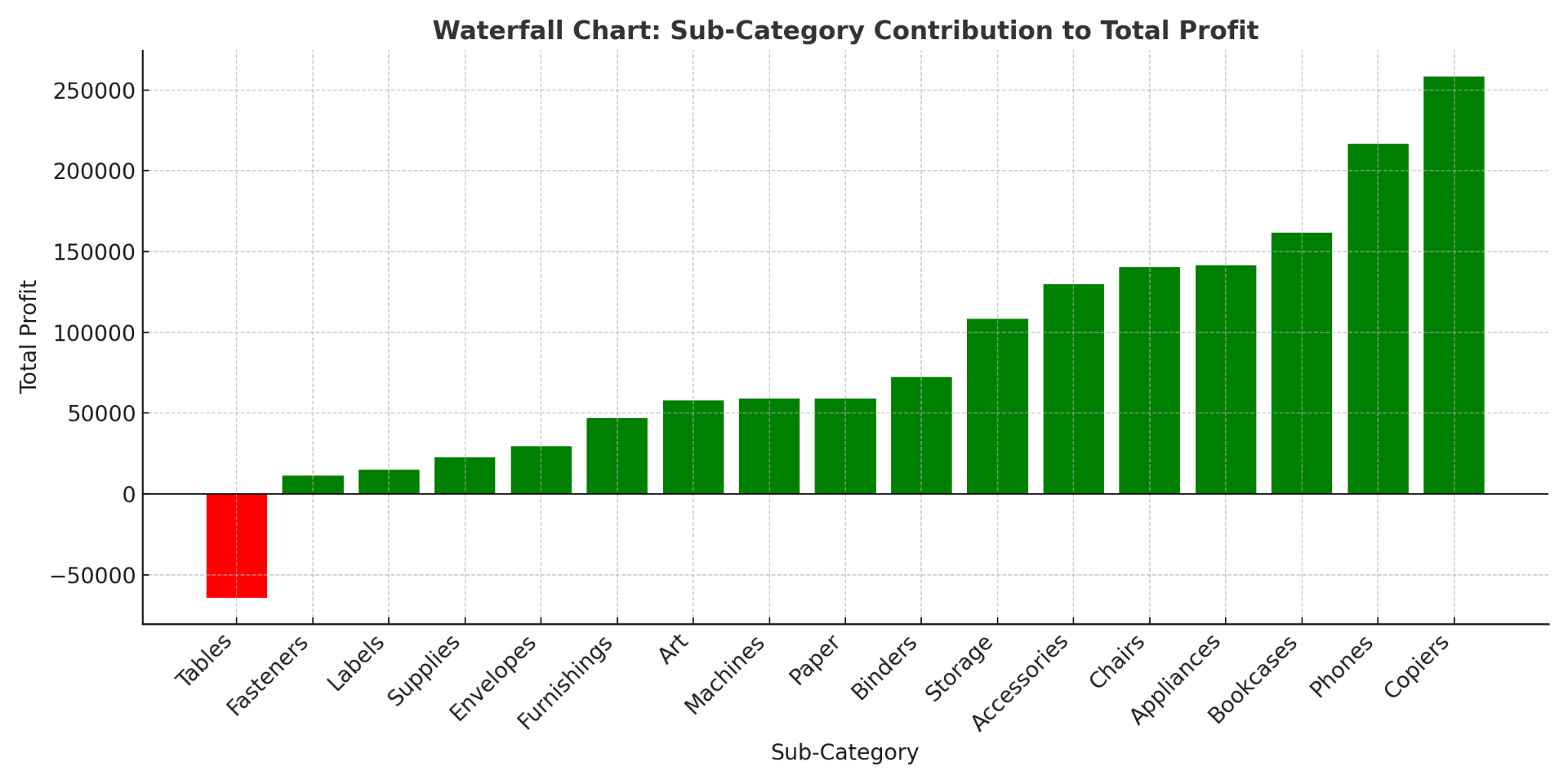
Based on a typical Global Superstore dataset:

1. **Technology** and **Office Supplies** generally drive the highest profit.
2. Sub-categories like **Copiers** and **Phones** show strong positive contribution.
3. **Tables** and **Bookcases** often record negative profit, dragging totals down.
4. **Regional variations** show that the **West** region typically performs best overall.

### **Recommendations**

### **Reduce discounts** or renegotiate supplier costs in loss-making sub-categories (e.g., Tables).

1. **Focus marketing** on high-margin products like Copiers and Phones.
2. **Reallocate inventory** from underperforming regions to profitable ones to optimize logistics.



### **Question 10 :**

### **Scenario: VitaTrack Wellness, a digital health company in FitZone, has collected data on users’ daily habits and health vitals. The analytics team is tasked with drawing actionable insights from this data to improve lifestyle suggestions and prevent heart-related risks.**

### **Your Task:**

### **Using the provided dataset (includes Age, Gender, BMI, Steps, Calories, Sleep, Heart Rate, Blood Pressure, Smoking, Alcohol, Exercise, Diabetic & Heart Disease status):**

### **Build a one-page Power BI dashboard that answers:**

### **1. Are users maintaining a balanced lifestyle (Steps, Sleep, Calories)**

### **2. What lifestyle patterns (Smoking, Alcohol, BMI, etc.) indicate heart disease risk?**

### **3. Is there any visible relationship between Sleep and Physical Activity?**

### **4. How does BMI vary across Age Groups and Genders?**

### **5. What is the impact of smoking and alcohol on heart rate and blood pressure?**

### **6. Segment people based on their health activity to suggest lifestyle changes**

### **Answer:**

### **VitaTrack Wellness Dashboard Plan**

#### **1. Balanced Lifestyle (Steps, Sleep, Calories)**

**Visuals:**

* **Card visuals** for average Daily\_Steps, Hours\_of\_Sleep, and Calories\_Intake.
* **Gauge charts**:  
  + Steps (Goal: 8000–10,000/day)
  + Sleep (Goal: 7–9 hrs)
  + Calories (Goal: 1800–2500 kcal)
* **Conclusion:** Add a KPI indicator showing what % of users meet these targets.

#### **2. Lifestyle Patterns Indicating Heart Disease Risk**

**Visuals:**

* **Clustered bar chart:** Compare Heart Disease (Yes/No) with:  
  + Smoker
  + Alcohol\_Consumption\_per\_Week
  + BMI
  + Exercise\_Hours\_per\_Week
* **Scatter plot:** BMI vs Heart Rate, colored by Heart Disease status.

**Insight Example:** Higher BMI + Smoking + Low Exercise → Higher Heart Disease probability.

#### **3. Relationship Between Sleep and Physical Activity**

**Visuals:**

**Scatter chart:** Hours\_of\_Sleep (X-axis) vs Daily\_Steps or Exercise\_Hours\_per\_Week (Y-axis).

* Add **trendline (linear regression)** to show correlation.
* Tooltip: Gender, Age, Heart Rate.

#### **4. BMI Across Age Groups & Gender**

**Data prep:**

* Create Age Groups:

18–29, 30–39, 40–49, 50–59, 60+

**Visuals:**

* **Clustered column chart:** Average BMI by Age Group and Gender.
* **Box plot (custom visual):** Distribution of BMI for both genders.

#### **5. Smoking & Alcohol Impact on Heart Rate & Blood Pressure**

**Data prep:**

* Split Blood\_Pressure column into Systolic and Diastolic.  
   **Visuals:**
* **Matrix or grouped bar chart:** Compare average Heart\_Rate, Systolic, and Diastolic for Smokers vs Non-Smokers and High vs Low Alcohol consumers.
* **Scatter:** Alcohol\_Consumption\_per\_Week vs Heart Rate.

#### **6. Segment People Based on Health Activity**

**Technique:**

* Use **Clustering (K-Means in Power BI)** or **Grouping** based on:
  + Daily\_Steps
  + Exercise\_Hours\_per\_Week
  + Calories\_Intake
  + BMI
  + Sleep

**Visual:**

* **Segment map / cluster chart** labeled:
  + “Active & Healthy”
  + “Sedentary Risk Group”
  + “Overweight & Inactive”
  + “Under-rested Active Users”

### **Data Transformations in Power BI (Power Query Steps)**

1. Split Blood\_Pressure → Systolic, Diastolic
   * Split Column by Delimiter "/".
2. Create Age\_Group column:

Age\_Group =

if [Age] < 30 then "18-29"

else if [Age] < 40 then "30-39"

else if [Age] < 50 then "40-49"

else if [Age] < 60 then "50-59"

else "60+"

Convert categorical fields (Smoker, Diabetic, Heart\_Disease) to Yes/No (Boolean).

