EDA(Exploration Data Analysis):

Step 1: Import Required Libraries

- Import essential libraries such as pandas for data manipulation, numpy for numerical computations, and visualization tools like matplotlib and seaborn.
- These libraries help in loading datasets, analyzing relationships, and visualizing patterns.

Step 2: Load the Datasets

- Read three datasets:
  - **Customers.csv**: Contains customer details (e.g., ID, Name, Region, Signup Date).
  - **Products.csv**: Contains product details (e.g., Product ID, Name, Category).
  - **Transactions.csv**: Records purchase transactions (e.g., Transaction ID, Customer ID, Product ID, Date).
- These datasets will be used for data analysis and clustering.

Step 3: Preview the Data

- Display the first few rows (head()) and last few rows (tail()) of each dataset to check the structure.
- This helps in understanding the data format and verifying if the files loaded correctly.

Step 4: Check Data Dimensions

- Use shape to check the number of rows and columns in each dataset.
- This provides an overview of dataset size before performing operations.

Step 5: Dataset Summary

- Use info() to get details about data types, missing values, and memory usage.
- Use describe() to view summary statistics like mean, standard deviation, minimum, and maximum values for numerical columns.
- Helps in detecting anomalies and understanding data distribution.

Step 6: Identify Missing Values

- Use isnull().sum() to count missing values in each column.
- If missing values exist, appropriate strategies like imputation or removal can be applied.

Step 7: Check Unique Values and Frequencies

- Use value_counts() to count unique values in categorical columns.
- This helps in understanding data distribution and identifying dominant categories.

Step 8: Encode Categorical Variables

- Convert categorical columns (e.g., Customer ID, Region, Product Name) into numeric form using **Label Encoding**.
- Encoding is necessary because machine learning models require numerical inputs.

Step 9: Compute Correlation Matrix

- Calculate the correlation between numerical features.
- Correlation helps identify relationships between variables and understand which factors might influence customer behavior.

Step 10: Visualize Correlation using Heatmaps

- Use **Seaborn's heatmap** to display correlations in a visually appealing format.
- Helps in detecting strong positive or negative relationships between features.

Step 11: Pairwise Data Visualization

- Use **Pairplots** to visualize relationships between multiple numerical features.
- This helps in identifying patterns, clusters, and trends in the dataset.

Business Insights:

1. Top-Selling Products

- Certain `ProductID`s appear frequently in transactions, suggesting they are in high demand.

- Understanding which products generate the most revenue can help optimize inventory management.

2. Customer Segmentation

- Some customers have multiple purchases, while others have a single transaction.

- Identifying high-value customers allows businesses to create loyalty programs and targeted promotions.

3. Sales Trends & Seasonality

- If TransactionDate data is recovered, a time-based analysis can identify peak sales periods.

- This insight can help businesses plan promotions and stock levels accordingly.

4. Pricing Strategy

- The dataset provides Price per product, which can help analyze pricing strategies.

- Comparing the sales performance of high-price vs. low-price products can inform future pricing decisions.

5. Revenue Growth Opportunities

- Customers who buy in bulk might be ideal candidates for special offers or business accounts.

- Identifying underperforming products can help phase out slow-moving stock and focus on high-performing ones.