

Exploratory Data Analysis

Insight-1: Total Number of Customers:

The total number of unique customers is calculated by using the `nunique()` method on the `CustomerID` column of the `customers.csv` file. This method counts all distinct customer IDs, which gives us the overall customer base.

`total_customers = customers['CustomerID'].nunique()`

Insight-2: Number of Products Sold:

To find the number of unique products sold, we use the `nunique()` method on the `ProductID` column from the `transactions.csv` file. This identifies how many different products are present in the transactions.

`unique_products_sold = transactions['ProductID'].nunique()`

Insight-3: Most Expensive Product:

By using the `idxmax()` method, we identify the product with the highest price from the `products.csv` file. The method returns the index of the row with the maximum price, which we use to retrieve the product details.

`most_expensive_product = products.loc[products['Price'].idxmax()]`

Insight-4 : Number of Orders per Customer:

Using the `groupby('CustomerID')` method on the `transactions.csv` file, we group the data by customer, and then use `size()` to count the number of transactions (orders) for each customer.

`orders_per_customer = transactions.groupby('CustomerID').size()`

Insight-5 : Customer Who Ordered the Maximum Number of Products:

To find the customer who placed the most orders, we identify the `CustomerID` associated with the maximum value in the `orders_per_customer` series. We use `orders_per_customer.max()` to get the maximum order count, then filter to get the corresponding customer.

`max_orders_customer = orders_per_customer[orders_per_customer == orders_per_customer.max()]`