CUSTOMER SHOPPING BEHAVIOR ANALYSIS

1. Project Overview

This project analyzes **customer shopping behavior** using transactional data from **3,900 customers** across multiple product categories. The objective is to uncover insights into **spending patterns, customer segmentation, product preferences, and subscription behavior** to support strategic decision-making and enhance business performance.

2. Dataset Summary

- **Total Records:** 3,900
- **Total Columns:** 18
- **Key Features:**
  - **Demographics:** Age, Gender, Location, Subscription Status
  - **Purchase Details:** Item Purchased, Category, Purchase Amount, Season, Size, Color
  - **Shopping Behavior:** Discount Applied, Promo Code Used, Previous Purchases, Frequency of Purchases, Review Rating, Shipping Type
- **Missing Data:** 37 missing values in the *Review Rating* column

3. Exploratory Data Analysis (Python)

The dataset was processed and analyzed using Python (pandas, NumPy, and matplotlib).

Data Preparation Steps

- **Data Loading:** Imported dataset using pandas.
- **Initial Exploration:** Used df.info() and df.describe() to inspect data structure and summary statistics.

```
#Reading the top 5 rows and structure of the data
df.head()
```

	Customer ID	Age	Gender	Item Purchased	Category	Purchase Amount (USD)	Location	Size	Color	Season	Review Rating	Subscription Status	Shipping Type	Discount Applied	Promo Code Used	Previous Purchases	Payment Method	Frequency of Purchases
0	1	55	Male	Blouse	Clothing	53	Kentucky	L	Gray	Winter	3.1	Yes	Express	Yes	Yes	14	Venmo	Fortnightly
1	2	19	Male	Sweater	Clothing	64	Maine	L	Maroon	Winter	3.1	Yes	Express	Yes	Yes	2	Cash	Fortnightly
2	3	50	Male	Jeans	Clothing	73	Massachusetts	S	Maroon	Spring	3.1	Yes	Free Shipping	Yes	Yes	23	Credit Card	Weekly
3	4	21	Male	Sandals	Footwear	90	Rhode Island	M	Maroon	Spring	3.5	Yes	Next Day Air	Yes	Yes	49	PayPal	Weekly
4	5	45	Male	Blouse	Clothing	49	Oregon	M	Turquoise	Spring	2.7	Yes	Free Shipping	Yes	Yes	31	PayPal	Annually

```
df.describe()  #It gives the statistics of the numeric columns
```

	Customer ID	Age	Purchase Amount (USD)	Review Rating	Previous Purchases
count	3900.000000	3900.000000	3900.000000	3863.000000	3900.000000
mean	1950.500000	44.068462	59.764359	3.750065	25.351538
std	1125.977353	15.207589	23.685392	0.716983	14.447125
min	1.000000	18.000000	20.000000	2.500000	1.000000
25%	975.750000	31.000000	39.000000	3.100000	13.000000
50%	1950.500000	44.000000	60.000000	3.800000	25.000000
75%	2925.250000	57.000000	81.000000	4.400000	38.000000
max	3900.000000	70.000000	100.000000	5.000000	50.000000

- **Missing Data Handling:** Imputed missing values in *Review Rating* using the **median rating** within each product by category.

```
#Replacing the null values with the median as mean can be affected by the outliers.
#Median is robust to outliers.

df['Review Rating'] = df.groupby('Category')['Review Rating'].transform(lambda x:x.fillna(x.median()))
```

```
df.isnull().sum() #Here null values in the Review Rating are replaced with the median and it is done groupby
```

```
Customer ID               0
Age                       0
Gender                    0
Item Purchased            0
Category                  0
Purchase Amount (USD)     0
Location                  0
Size                      0
Color                     0
Season                    0
Review Rating             0
Subscription Status       0
Shipping Type             0
Discount Applied          0
Promo Code Used           0
Previous Purchases        0
Payment Method            0
Frequency of Purchases    0
dtype: int64
```

- **Column Standardization:** Renamed columns to **snake_case** format for consistency.

```
#changing the column names into snake case
df.columns = df.columns.str.lower()
df.columns = df.columns.str.replace(' ','_')
df = df.rename (columns ={'purchase_amount_(usd)':'purchased_amount'})
```

```
df.columns #We can see all the column names are changed to lowercase
```

```
Index(['customer_id', 'age', 'gender', 'item_purchased', 'category',
       'purchased_amount', 'location', 'size', 'color', 'season',
       'review_rating', 'subscription_status', 'shipping_type',
       'discount_applied', 'promo_code_used', 'previous_purchases',
       'payment_method', 'frequency_of_purchases'],
      dtype='object')
```

- **Feature Engineering:**
    - Created **age_group** column by categorizing customer ages.
    - Derived **purchase_frequency_days** from transaction timestamps.
- **Data Consistency Check:** Verified redundancy between *discount_applied* and *promo_code_used*, and dropped the latter.
- **Database Integration:** Loaded the cleaned DataFrame into **PostgreSQL** for SQL-based business analysis.

```
#1st step: Connect to MySql


conn = mysql.connector.connect(
    host="localhost",      # or your host, e.g. "127.0.0.1"
    user="root",           # your MySQL username
    password="password",
    database="customer_behaviour"
)

'''if conn.is_connected():
    print("✅ Connection successful!")'''
database = "customer_behaviour"

engine = create_engine("mysql+pymysql://root:password@localhost/customer_behaviour")

#2nd Step : Load Dataframe into MySql
table_name = "customers"
df.to_sql(table_name, engine, if_exists="replace", index = False)

print(f"Data successfully loaded into table '{table_name} in database '{database}'.")

#pd.read_sql("Select * from customers limit 5;", engine)
```

```
Data successfully loaded into table 'customers in database 'customer_behaviour'.
```

4. Data Analysis using SQL (Business Transactions)

Structured SQL queries were used to answer key business questions:

1. **Revenue by Gender:** Compared total revenue generated by male vs. female customers.

```
5      -- Q1. What is the total revenue generated by male vs. female customers?
6      SELECT gender, SUM(purchased_amount) AS revenue
7      FROM customers
8      GROUP BY gender;
9
10
11
12
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

gender	revenue
Male	157890
Female	75191

2. **High-Spending Discount Users:** Identified customers using discounts who still spent above the average purchase amount.

customer_id	purchased_amount
2	64
3	73
4	90
7	85
9	97
12	68
13	72
16	81
20	90
22	62
24	88
29	94
32	79
33	67
35	91
37	69
40	60

3. **Top 5 Products by Rating:** Extracted products with the highest average review ratings.

item_purchased	Average Product Rating
Gloves	3.86
Sandals	3.84
Boots	3.82
Hat	3.80
Skirt	3.78

4. **Shipping Type Comparison:** Analyzed average purchase amounts between *Standard* and *Express* shipping.

shipping_type	Average Purchase Amount
Express	60.48
Standard	58.46

5. **Subscribers vs. Non-Subscribers:** Compared average spending and total revenue by subscription status.

subscription_status	total_customers	avg_spend	total_revenue
No	2847	59.87	170436
Yes	1053	59.49	62645

6. **Discount-Dependent Products:** Listed top 5 products with the highest percentage of discounted purchases.

item_purchased	discount_rate
Hat	50.0
Sneakers	49.66
Coat	49.07
Sweater	48.17
Pants	47.37

50.00

7. **Customer Segmentation:** Classified customers into *New*, *Returning*, and *Loyal* based on purchase history.

customer_segment	Number of Customers
Loyal	3116
Returning	701
New	83

8. **Top 3 Products per Category:** Identified the most purchased products within each category.

item_rank	category	item_purchased	total_orders
1	Accessories	Jewelry	171
2	Accessories	Sunglasses	161
3	Accessories	Belt	161
1	Clothing	Blouse	171
2	Clothing	Pants	171
3	Clothing	Shirt	169
1	Footwear	Sandals	160
2	Footwear	Shoes	150
3	Footwear	Sneakers	145
1	Outerwear	Jacket	163
2	Outerwear	Coat	161

9. **Repeat Buyers & Subscriptions:** Checked correlation between frequent buyers (>5 purchases) and subscription rates.

subscription_status	repeat_buyers
Yes	958
No	2518

10. **Revenue by Age Group:** Calculated total revenue contribution from each age segment.

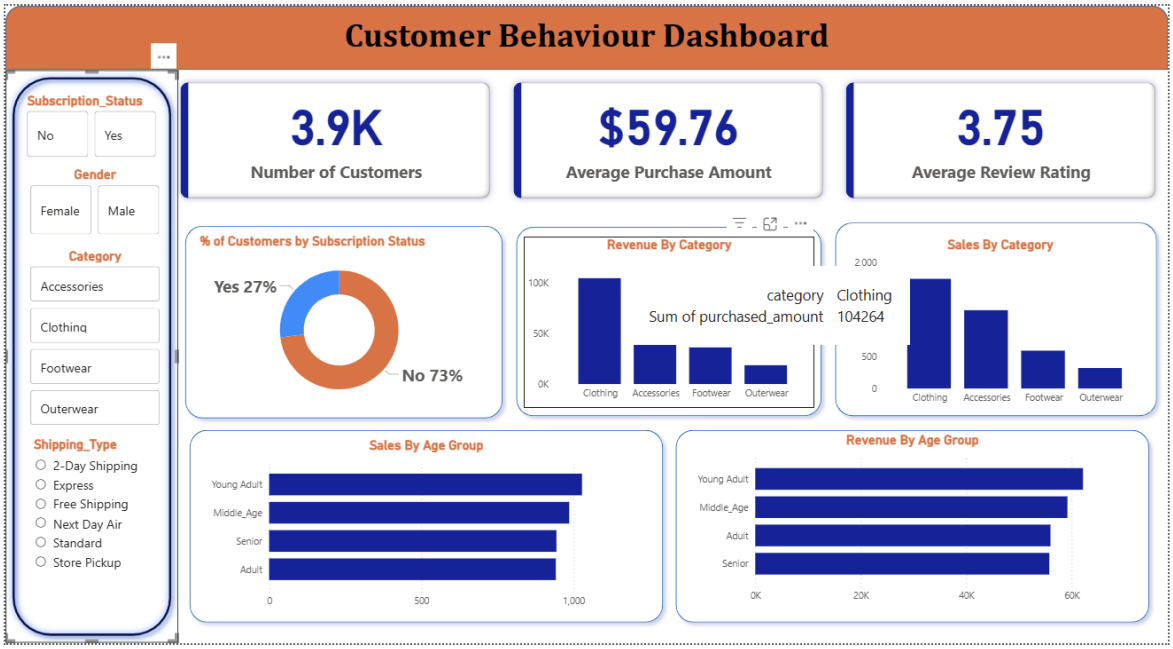
age_group	total_revenue
Young Adult	62143
Middle_Age	59197
Adult	55978
Senior	55763

5. Dashboard in Power BI

An **interactive Power BI dashboard** was developed to visualize customer insights, including:

- Revenue distribution by gender and age group
- Purchase trends by category and season
- Subscriber vs. non-subscriber comparisons

- Discount utilization and high-value customer segments
- Product ratings and shipping type performance



6. Business Recommendations

Based on data insights, the following strategic recommendations were made:

- **Boost Subscriptions:** Promote exclusive benefits and personalized offers to increase subscriber base.
- **Loyalty Programs:** Introduce reward systems to retain and incentivize frequent buyers.
- **Optimize Discount Policies:** Balance promotional discounts with profit margins to sustain revenue.
- **Product Positioning:** Highlight top-rated and best-selling products in marketing campaigns.
- **Targeted Marketing:** Focus marketing efforts on high-revenue age groups and customers preferring express shipping.