Customer Shopping Behavior Analysis

1. Project Overview

This project analyzes customer shopping behavior using transactional data from 3,900 purchases across various product categories. The goal is to uncover insights into spending patterns, customer segments, product preferences, and subscription behavior to guide strategic business decisions.

2. Dataset Summary

- Rows: 3,900
- Columns: 18
- Key Features:
  - Customer 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 values in Review Rating column

3. Exploratory Data Analysis using Python

We began with data preparation and cleaning in Python:

- **Data Loading:** Imported the dataset using `pandas`.
- **Initial Exploration:** Used `df.info()` to check structure and `.describe()` for summary statistics.

```
var.info()
print(var)
```

Python

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3900 entries, 0 to 3899
Data columns (total 18 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   customer_id             3900 non-null   int64
 1   age                     3900 non-null   int64
 2   gender                  3900 non-null   object
 3   item_purchased          3900 non-null   object
 4   category                3900 non-null   object
 5   purchase_amount_(usd)   3900 non-null   int64
 6   location                3900 non-null   object
 7   size                    3900 non-null   object
 8   color                   3900 non-null   object
 9   season                  3900 non-null   object
 10  review_rating           3900 non-null   float64
 11  subscription_status     3900 non-null   object
 12  shipping_type           3900 non-null   object
 13  discount_applied        3900 non-null   object
 14  promo_code_used         3900 non-null   object
 15  previous_purchases      3900 non-null   int64
 16  payment_method          3900 non-null   object
 17  frequency_of_purchases  3900 non-null   object
dtypes: float64(1), int64(4), object(13)
memory usage: 548.6+ KB
...
3898             No                24          Venmo              Weekly
3899             No                33          Venmo           Quarterly

[3900 rows x 18 columns]
```

```
var.describe(include='all')
#it gives a summary of categorial columns and numeriscal columns
```

Python

	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	Frequenc o Purchase
count	3900.000000	3900.000000	3900	3900	3900	3900.000000	3900	3900	3900	3900	3863.000000	3900	3900	3900	3900	3900.000000	3900	390
unique	NaN	NaN	2	25	4	NaN	50	4	25	4	NaN	2	6	2	2	NaN	6	
top	NaN	NaN	Male	Blouse	Clothing	NaN	Montana	M	Olive	Spring	NaN	No	Free Shipping	No	No	NaN	PayPal	Every Month
freq	NaN	NaN	2652	171	1737	NaN	96	1755	177	999	NaN	2847	675	2223	2223	NaN	677	58
mean	1950.500000	44.068462	NaN	NaN	NaN	59.764359	NaN	NaN	NaN	NaN	3.750065	NaN	NaN	NaN	NaN	25.351538	NaN	Nal
std	1125.977353	15.207589	NaN	NaN	NaN	23.685392	NaN	NaN	NaN	NaN	0.716983	NaN	NaN	NaN	NaN	14.447125	NaN	Nal
min	1.000000	18.000000	NaN	NaN	NaN	20.000000	NaN	NaN	NaN	NaN	2.500000	NaN	NaN	NaN	NaN	1.000000	NaN	Nal
25%	975.750000	31.000000	NaN	NaN	NaN	39.000000	NaN	NaN	NaN	NaN	3.100000	NaN	NaN	NaN	NaN	13.000000	NaN	Nal
50%	1950.500000	44.000000	NaN	NaN	NaN	60.000000	NaN	NaN	NaN	NaN	3.800000	NaN	NaN	NaN	NaN	25.000000	NaN	Nal
75%	2925.250000	57.000000	NaN	NaN	NaN	81.000000	NaN	NaN	NaN	NaN	4.400000	NaN	NaN	NaN	NaN	38.000000	NaN	Nal
max	3900.000000	70.000000	NaN	NaN	NaN	100.000000	NaN	NaN	NaN	NaN	5.000000	NaN	NaN	NaN	NaN	50.000000	NaN	Nal

- **Missing Data Handling:** Checked for null values and imputed missing values in the `Review Rating` column using the median rating of each product category.

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

```
var.isnull().sum()
print(var)
```

```
     Customer ID  Age  Gender Item Purchased     Category  \
0              1   55    Male         Blouse     Clothing
1              2   19    Male        Sweater     Clothing
2              3   50    Male          Jeans     Clothing
3              4   21    Male        Sandals     Footwear
4              5   45    Male         Blouse     Clothing
...          ...  ...     ...            ...          ...
3895        3896   40  Female         Hoodie     Clothing
3896        3897   52  Female       Backpack  Accessories
3897        3898   46  Female           Belt  Accessories
3898        3899   44  Female          Shoes     Footwear
3899        3900   52  Female        Handbag  Accessories

     Purchase Amount (USD)       Location Size      Color  Season  \
0                       53       Kentucky    L       Gray  Winter
1                       64          Maine    L     Maroon  Winter
2                       73  Massachusetts    S     Maroon  Spring
3                       90   Rhode Island    M     Maroon  Spring
4                       49         Oregon    M  Turquoise  Spring
...                    ...            ...  ...        ...     ...
3895                    28       Virginia    L  Turquoise  Summer
```

- **Column Standardization:** Renamed columns to **snake case** for better readability and documentation.

```
var.columns=var.columns.str.lower()
var.columns=var.columns.str.replace(' ','_')
print(var)
```

```
      customer_id  age  gender item_purchased     category  \
0               1   55    Male         Blouse     Clothing
1               2   19    Male        Sweater     Clothing
2               3   50    Male          Jeans     Clothing
3               4   21    Male        Sandals     Footwear
4               5   45    Male         Blouse     Clothing
...           ...  ...     ...            ...          ...
3895         3896   40  Female         Hoodie     Clothing
3896         3897   52  Female       Backpack  Accessories
3897         3898   46  Female           Belt  Accessories
3898         3899   44  Female          Shoes     Footwear
3899         3900   52  Female        Handbag  Accessories

      purchase_amount_(usd)       location size       color  season  \
0                        53       Kentucky    L        Gray  Winter
1                        64          Maine    L      Maroon  Winter
2                        73  Massachusetts    S      Maroon  Spring
3                        90   Rhode Island    M      Maroon  Spring
4                        49         Oregon    M   Turquoise  Spring
...                     ...            ...  ...         ...     ...
3895                     28       Virginia    L   Turquoise  Summer
3896                     49           Iowa    L       White  Spring
3897                     33     New Jersey    L       Green  Spring
3898                     77      Minnesota    S       Brown  Summer
3899                     81     California    M       Beige  Spring
...
```

```
class 'pandas.core.frame.DataFrame'>
angeIndex: 3900 entries, 0 to 3899
ata columns (total 18 columns):
#   Column                  Non-Null Count  Dtype
--  ------                  --------------  -----
0   customer_id             3900 non-null   int64
1   age                     3900 non-null   int64
2   gender                  3900 non-null   object
3   item_purchased          3900 non-null   object
4   category                3900 non-null   object
5   purchase_amount_(usd)   3900 non-null   int64
6   location                3900 non-null   object
7   size                    3900 non-null   object
8   color                   3900 non-null   object
9   season                  3900 non-null   object
10  review_rating           3900 non-null   float64
11  subscription_status     3900 non-null   object
12  shipping_type           3900 non-null   object
13  discount_applied        3900 non-null   object
14  promo_code_used         3900 non-null   object
15  previous_purchases      3900 non-null   int64
16  payment_method          3900 non-null   object
17  frequency_of_purchases  3900 non-null   object
types: float64(1), int64(4), object(13)
emory usage: 548.6+ KB
```

- **Feature Engineering:**
    - Created **age_group** column by binning customer ages.
    - Created **purchase_frequency_days** column from purchase data.

```
#create a column age_group
labels=['young','adult','midage','senior']
var['age_group']=pd.qcut(var['age'],q=4,labels=labels)
print(var)
```

```
#purchase frequence days column

frequency_mapping ={
    'Fortnightly':14,
    'Weekly':7,
    'Monthly':30,
    'Quarterly':90,
    'Bi-Weekly':14,
    'Annually':365,
    'Every 3 Months':90
}

var['purchases_frequency_days']=var['frequency_of_purchases'].map(frequency_mapping)
print(var)
```

- **Data Consistency Check:** Verified if `discount_applied` and `promo_code_used` were redundant; dropped `promo_code_used`.

```
var=var.drop('promo_code_used',axis=1)
print(var)
```

- **Database Integration:** Connected Python script to MySQL and loaded the cleaned DataFrame into the database for SQL analysis.

```python
from sqlalchemy import create_engine
import urllib.parse

# 👉 USE YOUR ALREADY MODIFIED DATAFRAME (df)

# MySQL connection
password = urllib.parse.quote_plus("Pooja@55")

engine = create_engine(
    f"mysql+mysqlconnector://root:{password}@localhost:3306/map"
)

# Upload modified dataframe
var.to_sql(
    name="customer_modified",
    con=engine,
    if_exists="replace",
    index=False
)

print("Modified Pandas DataFrame uploaded successfully!")
```

```
Modified Pandas DataFrame uploaded successfully!
```

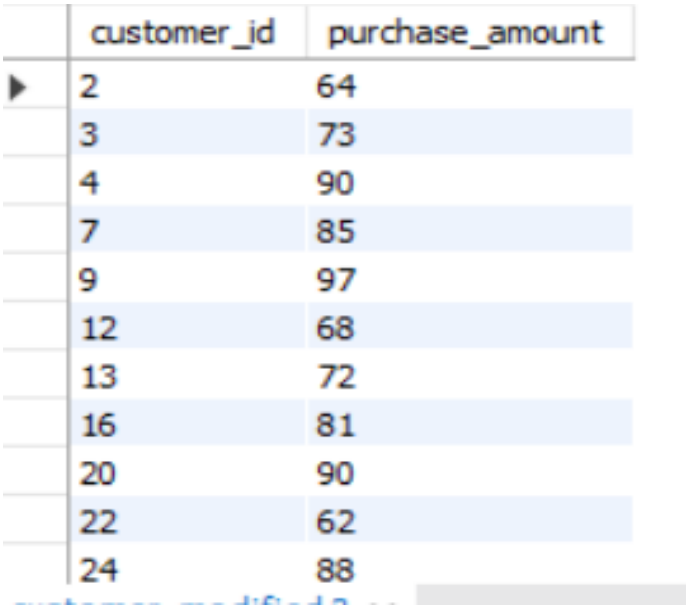
4. Data Analysis using MySQL (Business Transactions)

We performed structured analysis in PostgreSQL to answer key business questions:

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

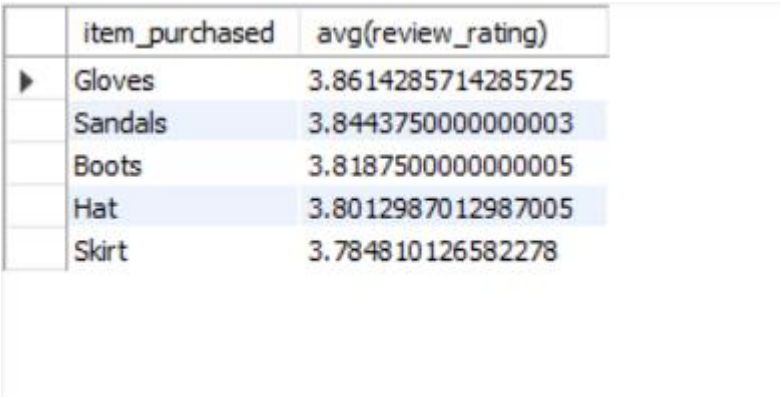
	gender	revenue
▶	Male	157890
	Female	75191

2. **High-Spending Discount Users** – Identified customers who used discounts but still spent above the average purchase amount.



customer_id	purchase_amount
2	64
3	73
4	90
7	85
9	97
12	68
13	72
16	81
20	90
22	62
24	88

3. **Top 5 Products by Rating** – Found products with the highest average review ratings.



item_purchased	avg(review_rating)
Gloves	3.8614285714285725
Sandals	3.8443750000000003
Boots	3.8187500000000005
Hat	3.8012987012987005
Skirt	3.784810126582278

4. **Shipping Type Comparison** – Compared average purchase amounts between Standard and Express shipping.

shipping_type	avg(purchase_amount)
Express	60.4752
Standard	58.4602

5. **Subscribers vs. Non-Subscribers** – Compared average spend and total revenue across subscription status.

subscription_status	total_customers	avg_spend	total_revenue
Yes	1053	59.4919	62645
No	2847	59.8651	170436

6. **Discount-Dependent Products** – Identified 5 products with the highest percentage of discounted purchases.

item_purchased	discount_percentage
Hat	50.0000
Sneakers	49.6552
Coat	49.0683
Sweater	48.1707
Pants	47.3684

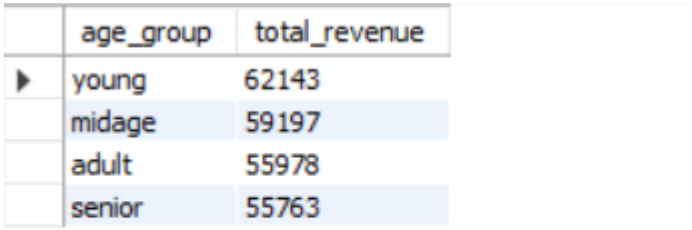
7. **Customer Segmentation** – Classified customers into New, Returning, and Loyal segments based on purchase history.

customer_segment	Number of Customers
Loyal	3116
Returning	701
New	83

8. **Repeat Buyers & Subscriptions** – Checked whether customers with >5 purchases are more likely to subscribe.

subscription_status	repeat_buyers
Yes	958
No	2518

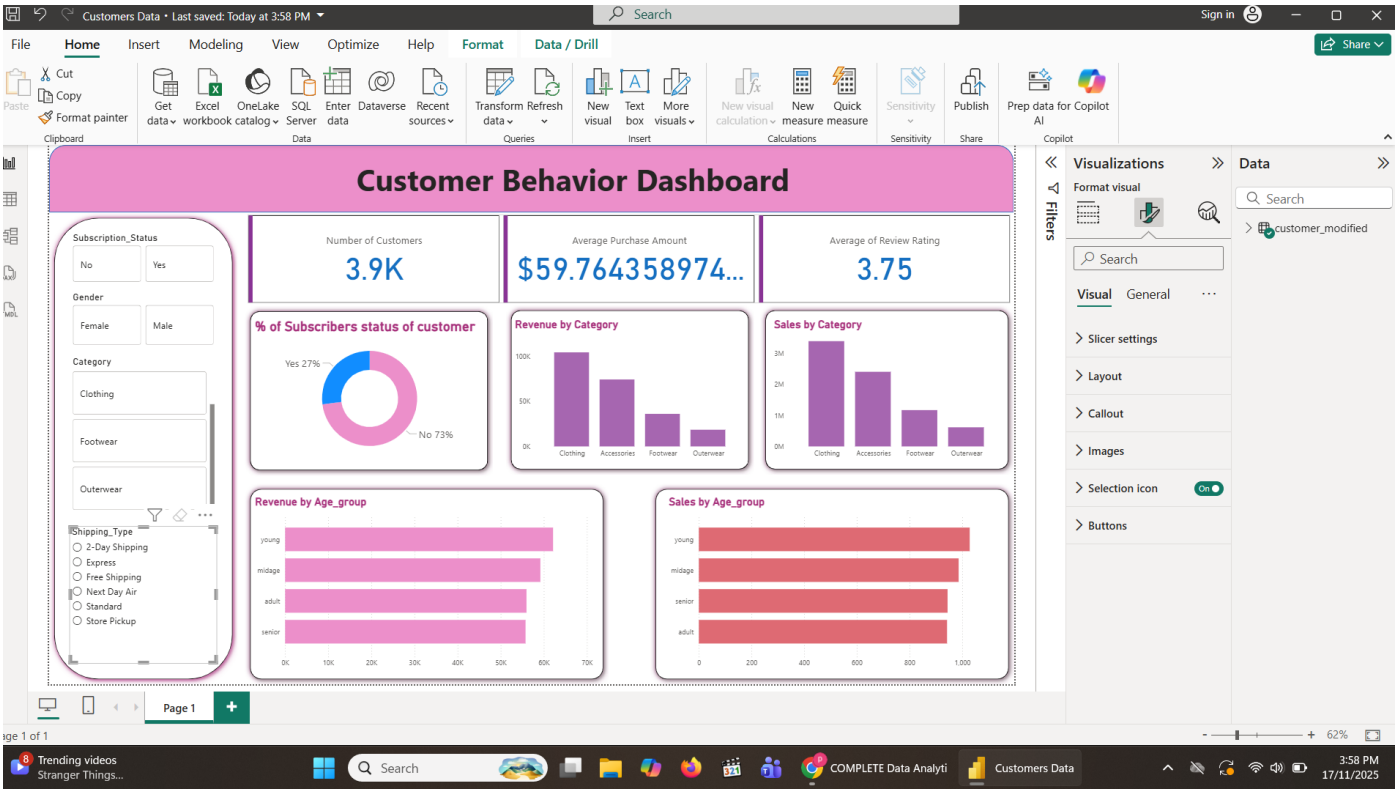
9.Revenue by Age Group – Calculated total revenue contribution of each age group.



age_group	total_revenue
young	62143
midage	59197
adult	55978
senior	55763

5. Dashboard in Power BI

Finally, we built an interactive dashboard in **Power BI** to present insights visually.



6. Business Recommendations

- **Boost Subscriptions** – Promote exclusive benefits for subscribers.
- **Customer Loyalty Programs** – Reward repeat buyers to move them into the “Loyal” segment.
- **Review Discount Policy** – Balance sales boosts with margin control.
- **Product Positioning** – Highlight top-rated and best-selling products in campaigns.
- **Targeted Marketing** – Focus efforts on high-revenue age groups and express-shipping users.