Customer Shopping Behavior Analysis

1.Project Overview

This project focuses on analyzing customer purchasing patterns to extract actionable business insights. The analysis integrates Python, MySQL, and Power BI, forming an end-to-end data pipeline including data cleaning, database querying and visualization.

The dataset contains 3,900 customer transactions across 18 features, enabling analysis across demographics, product categories, subscription behavior, and revenue contribution.

This project helps businesses understand customer behavior, identify profitable segments, and make data-driven decisions.

2. Dataset Summary

Attribute	Description
Rows	3,900
Columns	18
Key Features	Age, Gender, Purchase Amount, Category, Shipping Type, Discount Applied, Review Rating, Previous Purchases, Subscription Status.
Missing Values	37 Null values in Review Rating column

3. Python Workflow

The initial stage of the project was completed using Jupyter Notebook for Exploratory Data Analysis and Data Cleaning.

Key steps performed

Loaded dataset using pandas.read_csv()

Basic structure check using .info()

```
df.info()
```

```
<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           3863 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
```

Use .describe()

```
df.describe(include='all')
```

	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	3
count	3900.000000	3900.000000	3900	3900	3900	3900.000000	3900	3900	3900	3900	3863.000000	3900	3900	3900	3900	
unique	NaN	NaN	2	25	4	NaN	50	4	25	4	NaN	2	6	2	2	
top	NaN	NaN	Male	Blouse	Clothing	NaN	Montana	M	Olive	Spring	NaN	No	Free Shipping	No	No	
freq	NaN	NaN	2652	171	1737	NaN	96	1755	177	999	NaN	2847	675	2223	2223	
mean	1950.500000	44.068462	NaN	NaN	NaN	59.764359	NaN	NaN	NaN	NaN	3.750065	NaN	NaN	NaN	NaN	
std	1125.977353	15.207589	NaN	NaN	NaN	23.685392	NaN	NaN	NaN	NaN	0.716983	NaN	NaN	NaN	NaN	
min	1.000000	18.000000	NaN	NaN	NaN	20.000000	NaN	NaN	NaN	NaN	2.500000	NaN	NaN	NaN	NaN	
25%	975.750000	31.000000	NaN	NaN	NaN	39.000000	NaN	NaN	NaN	NaN	3.100000	NaN	NaN	NaN	NaN	
50%	1950.500000	44.000000	NaN	NaN	NaN	60.000000	NaN	NaN	NaN	NaN	3.800000	NaN	NaN	NaN	NaN	
75%	2925.250000	57.000000	NaN	NaN	NaN	81.000000	NaN	NaN	NaN	NaN	4.400000	NaN	NaN	NaN	NaN	
max	3900.000000	70.000000	NaN	NaN	NaN	100.000000	NaN	NaN	NaN	NaN	5.000000	NaN	NaN	NaN	NaN	

Missing values identification using df.isna().sum()

```
# Checking for null values
df.isnull().sum()
```

```
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             37
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
```

Imputed missing values in review_rating column using median per category

Removed redundant column promo_code_used

Feature engineering:

Created Age Group column (Young Adult, Middle-aged, Adult, Senior),

```
group = ['Young Adult', 'Adult', 'Mid Age', 'Senior Age']
df['age_group'] = pd.qcut(df['age'], q=4, labels = group)
```

```
df[['age','age_group']].head(7)
```

	age	age_group
0	55	Mid Age
1	19	Young Adult
2	50	Mid Age
3	21	Young Adult
4	45	Mid Age
5	46	Mid Age
6	63	Senior Age

Created purchase_frequency_days,

Connected Python to MySQL and imported cleaned dataset to database for SQL analysis,

```
from sqlalchemy import create_engine
import pymysql
import pandas as pd

username = "root"
password = "%40**********"  # encode @ as %40
host = "localhost"
port = "3306"
database = "customer_Behaviour"

engine = create_engine(
    f"mysql+pymysql://{username}:{password}@{host}:{port}/{database}"
)

df.to_sql("customer", engine, if_exists="replace", index=False)

pd.read_sql("SELECT * FROM customer LIMIT 5;", engine)
```

4. SQL Analysis

The cleaned data was loaded into a MySQL database and queries were executed to answer critical business questions. Some examples include:

Revenue generated by gender :

```
select gender, sum(purchase_amount) as Revenue
from customer
group by gender;
```

gender	Revenue
Male	157890
Female	75191

Customers who used a discount but spent above average :

```
select customer_id, purchase_amount
from customer
where discount_applied = "yes" and purchase_amount >= (select avg(purchase_amount) from customer);
```

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

Top 5 Products by Review Rating :

```
select item_purchased, round(avg(review_rating),2) as "avg rating"
from customer
group by item_purchased
order by avg(review_rating) desc limit 5;
```

item_purchased	avg rating
Gloves	3.86
Sandals	3.84
Boots	3.82
Hat	3.8
Skirt	3.78

Customer Segmentation (New / Returning / Loyal) :

```
with customer_type as (
SELECT customer_id, previous_purchases,
CASE
    WHEN previous_purchases = 1 THEN 'New'
    WHEN previous_purchases BETWEEN 2 AND 10 THEN 'Returning'
    ELSE 'Loyal'
    END AS customer_segment
FROM customer)
select customer_segment,count(*) AS "Number of Customers"
from customer_type
group by customer_segment;
```

customer_segment	Number of Customers
Loyal	3116
Returning	701
New	83

5. Power BI Dashboard

After completing SQL analytics, results were connected into Power BI using Database connector and designed an interactive analytical dashboard that included slicers, filters, KPIs, bar charts and donut charts.

Power BI Dashboard KPIs :

- Total Customers
- Average Purchase Amount
- Average Review Rating
- Revenue by Category / Age Group
- Sales by Category
- % Customers by Subscription Status

Power BI DAX Measures Used :

Measure Name	DAX Formula
Total Customers	Total Customers = COUNT(customer[customer_id])
Average Purchase Amount	Avg Purchase Amount = AVERAGE(customer[purchase_amount])
Average Review Rating	Avg Review Rating = AVERAGE(customer[review_rating])
Total Revenue	Total Revenue = SUM(customer[purchase_amount])
Subscription %	Subscription % = DIVIDE(CALCULATE(COUNT(customer[customer_id]), customer[subscription_status] = "Yes"), COUNT(customer[customer_id]))
Revenue by Category	Revenue by Category = SUM(customer[purchase_amount])
Revenue by Age Group	Revenue by Age Group = SUM(customer[purchase_amount])

Customer Segment Analysis :

Visualization	Purpose
Clustered Bar Chart	Segment wise customer count
Donut chart	Returning vs Loyal customers distribution
Drill-through	Customer-wise order behavior

6. Insights & Interpretation

Key Insights

Young Adults contribute the highest revenue, followed by Middle-aged customers,

Express shipping customers spend more than standard shipping customers,

Top-rated products include Gloves, Sandals, Boots, Hat & Skirt,

Subscription customers show higher transaction counts, but non-subscribers generate larger total revenue due to volume,

Accessories and Clothing are the most purchased categories,

Significant portion of revenue driven by Returning customers segment.

7. Business Recommendations

- Promote subscription benefits to increase conversion
- Introduce loyalty rewards for returning customers

- Highlight best rated and top selling products
- Run high-value marketing campaigns around young adults

Conclusion

This project demonstrates the power of integrating Python, SQL, and Power BI into a unified analytics workflow. Each tool played a strategic role: Python for cleaning, SQL for structured analysis, and Power BI for visualization. The insights derived can support targeted marketing, product placement, and business decision-making.

Tool / Platform	Purpose
MySQL Workbench	Data storage, structured query processing
Python (Jupyter Notebook)	Data cleaning and preprocessing
Power BI Desktop	Data modeling, DAX measures, visualization
Power Query M	ETL, transformations
Power BI Bookmarks & Drillthrough	Interactive navigation & detailed insights