Customer Shopping Behaviour Analysis

1. Project Overview

This project analyses customer shopping behaviour using transactional data from 3,900 purchases across multiple product categories. The analysis focuses on identifying spending patterns, customer segments, product preferences, and subscription behaviour to support data-driven business decisions using Python, SQL Server, and Power BI.

2. Data 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 behaviour (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 & Data Preparation 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

```
[23]  df.describe()
      # check statistics of numerical 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**: Checked for null values and imputed missing values in the Review Rating column using the median rating of each product category.
- **Column Standardization**: Renamed columns to **snake case** for better readability and documentation.
- **Feature Engineering**:
  - Created **age_group** column by binning customer ages.
  - Created **purchase_frequency_days** column from purchase data.

- **Data Consistency Check**: Verified if discount_applied and promo_code_used were redundant; dropped promo_code_used.
- **Data Export & Database Loading**: Exported the cleaned and feature-engineered dataset to CSV format and imported it into Microsoft SQL Server for structured SQL analysis.

4. Data Analysis using SQL (Business Transactions)

We performed structured analysis in Microsoft SQL Server to answer key business questions:

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

```sql
-- Q1. what is total revenue generated by male vs female customers? --
select gender, sum(purchase_amount) as revenue
from Customer_data
group by gender;
```

No issues found

Results | Messages

gender	revenue
Male	157890
Female	75191

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

```sql
--Q2. Which customers used a discount but still spent more than the average purchase amount? --
select customer_id, purchase_amount
from Customer_data
where discount_applied ='Yes' and
purchase_amount >=(select avg(purchase_amount) from Customer_data);
```

100 % No issues found

Results | Messages

	customer_id	purchase_amount
1	2	64
2	3	73
3	4	90
4	7	85
5	9	97
6	12	68
7	13	72
8	16	81
9	20	90
10	22	62
11	24	88
12	29	94
13	32	79
14	33	67
15	35	91
16	37	69
17	40	60
18	41	76

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

```sql
-- Q3. Which are the top 5 products with the highest average review rating? --
select top 5 item_purchased, round(avg(review_rating),2) as avg_rating
from Customer_data
group by item_purchased
order by avg_rating desc;
```

100 % No issues found

Results | Messages

	item_purchased	avg_rating
1	Gloves	3.86
2	Sandals	3.84
3	Boots	3.82
4	Hat	3.8
5	T-shirt	3.78

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

```sql
--Q4. Compare the average Purchase Amounts between Standard and Express Shipping.  --
select shipping_type, avg(purchase_amount) as avg_purchase_amout
from Customer_data
where shipping_type in('Express','Standard')
group by shipping_type
order by avg_purchase_amout desc;
```

100 % No issues found

Results Messages

	shipping_type	avg_purchase_amout
1	Express	60
2	Standard	58

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

```sql
--Q5. Do subscribed customers spend more? Compare average spend and total revenue between subscribers and non-subscribers. --
select subscription_status, count(customer_id) as total_customer,
avg(purchase_amount) as avg_spend,
sum(purchase_amount) as total_revenue
from Customer_data
group by subscription_status
order by total_revenue, avg_spend desc;
```

100 % No issues found

Results Messages

	subscription_status	total_customer	avg_spend	total_revenue
1	Yes	1053	59	62645
2	No	2847	59	170436

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

```sql
--Q6. Which 5 products have the highest percentage of purchases with discounts applied? --
select top 5
item_purchased,
round(100* sum(case when discount_applied = 'Yes' then 1 else 0 end)/count(*), 2) as discount_rate
from Customer_data
group by item_purchased
order by discount_rate desc;
```

100 % No issues found

Results Messages

	item_purchased	discount_rate
1	Hat	50
2	Coat	49
3	Sneakers	49
4	Sweater	48
5	Pants	47

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

```sql
--Q7. Segment customers into New, Returning, and Loyal based on their total number of previous purchases, and show the count of each segment. --
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_data
)
select customer_segment, count(*) as no_of_customers
from customer_type
group by customer_segment;
```

100 % No issues found

Results Messages

	customer_segment	no_of_customers
1	new	83
2	returning	701
3	loyal	3116

H. **Top 3 Products per Category** – Listed the most purchased products within each category.

```sql
--Q8. What are the top 3 most purchased products within each category?  --
with item_counts as (
select category, item_purchased,
count(customer_id) as total_orders,
row_number() over(partition by category order by count(customer_id) desc) as item_rank
from Customer_data
group by category, item_purchased
)
select item_rank, category, item_purchased, total_orders
from item_counts
where item_rank <=3;
```

100 % — No issues found

Results | Messages

	item_rank	category	item_purchased	total_orders
1	1	Accessories	Jewelry	171
2	2	Accessories	Sunglasses	161
3	3	Accessories	Belt	161
4	1	Clothing	Blouse	171
5	2	Clothing	Pants	171
6	3	Clothing	Shirt	169
7	1	Footwear	Sandals	160
8	2	Footwear	Shoes	150
9	3	Footwear	Sneakers	145
10	1	Outerwear	Jacket	163
11	2	Outerwear	Coat	161

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

```sql
--Q9. Are customers who are repeat buyers (more than 5 previous purchases) also likely to subscribe? --
select subscription_status,
count(customer_id) as repeat_buyers
from Customer_data
where previous_purchases > 5
group by subscription_status;
```

100 % — No issues found

Results | Messages

	subscription_status	repeat_buyers
1	Yes	958
2	No	2518

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

```sql
--Q10. What is the revenue contribution of each age group? --
select age_group,
sum(purchase_amount) as total_revenue
from Customer_data
group by age_group
order by total_revenue desc;
```

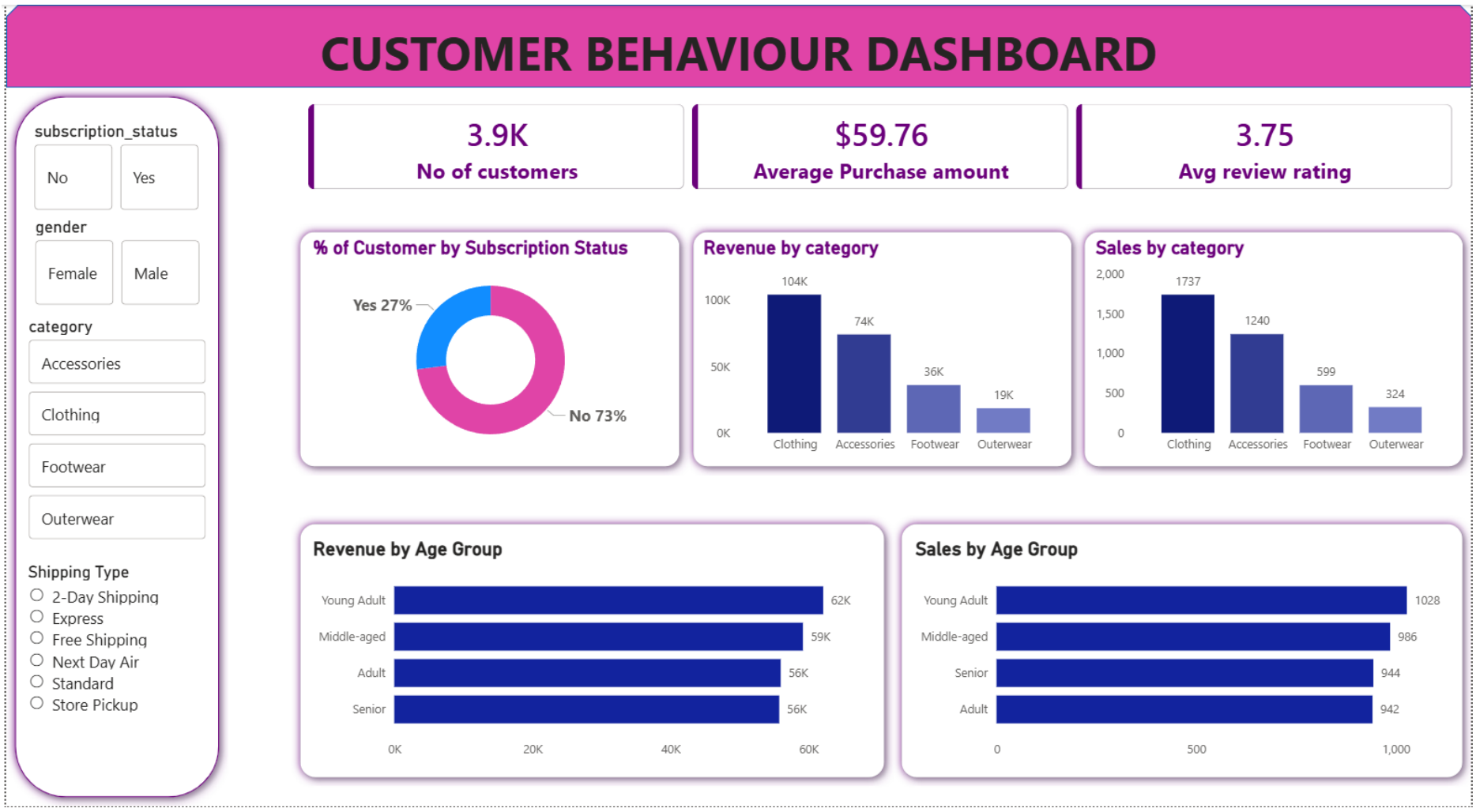
100 % — No issues found

Results | Messages

	age_group	total_revenue
1	Young Adult	62143
2	Middle-aged	59197
3	Adult	55978
4	Senior	55763

5. Analytical Report in Power BI

Finally, an interactive analytical report was built in Power BI to visualize insights from the SQL analysis and support business decision-making.



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.

7. Tools & Technologies Used

- Python (Pandas, NumPy) – Data cleaning and feature engineering
- Microsoft SQL Server – Structured data analysis using SQL
- Power BI – Analytical reporting and visualization