Project Report: Customer Shopping Behaviour Analysis

1. Project Overview

The primary goal of this project is to analyze customer shopping trends and behaviors to gain actionable insights for retail strategy. By examining variables such as purchase frequency, item categories, and payment methods, this analysis aims to identify key demographics and preferences that drive sales.

2. Dataset Summary

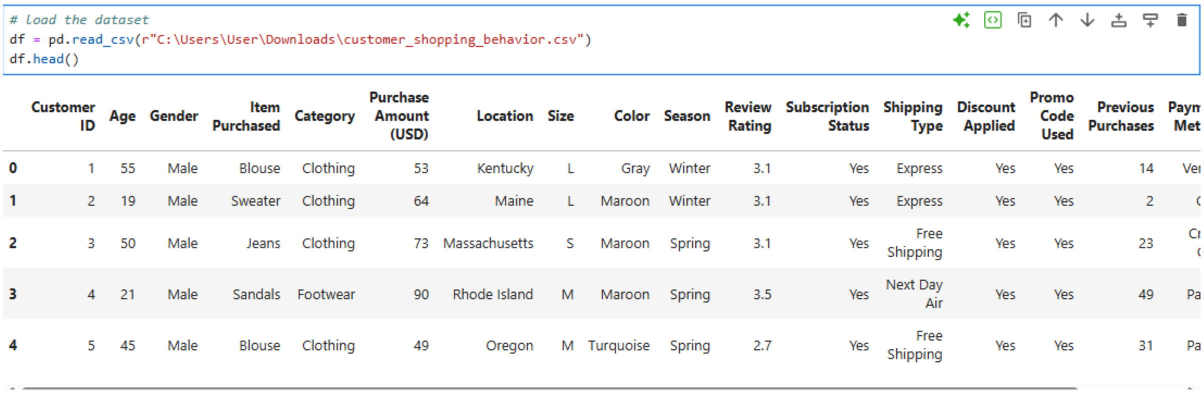
The dataset, customer_shopping_behavior.csv, contains information on 3,900 customer transactions. It includes 18 columns covering demographic data, purchase details, and customer loyalty metrics.

Key Features Include:

- **Customer Demographics:** Age, Gender, and Location.
- **Transaction Details:** Item Purchased, Category, Purchase Amount (USD), and Size.
- **Purchase Context:** Color, Season, and Shipping Type.
- **Customer Feedback & Loyalty:** Review Rating, Subscription Status, and Frequency of Purchases.
- **Payment & Promotions:** Payment Method, Discount Applied, and Promo Code Used.

3. EDA Using Python

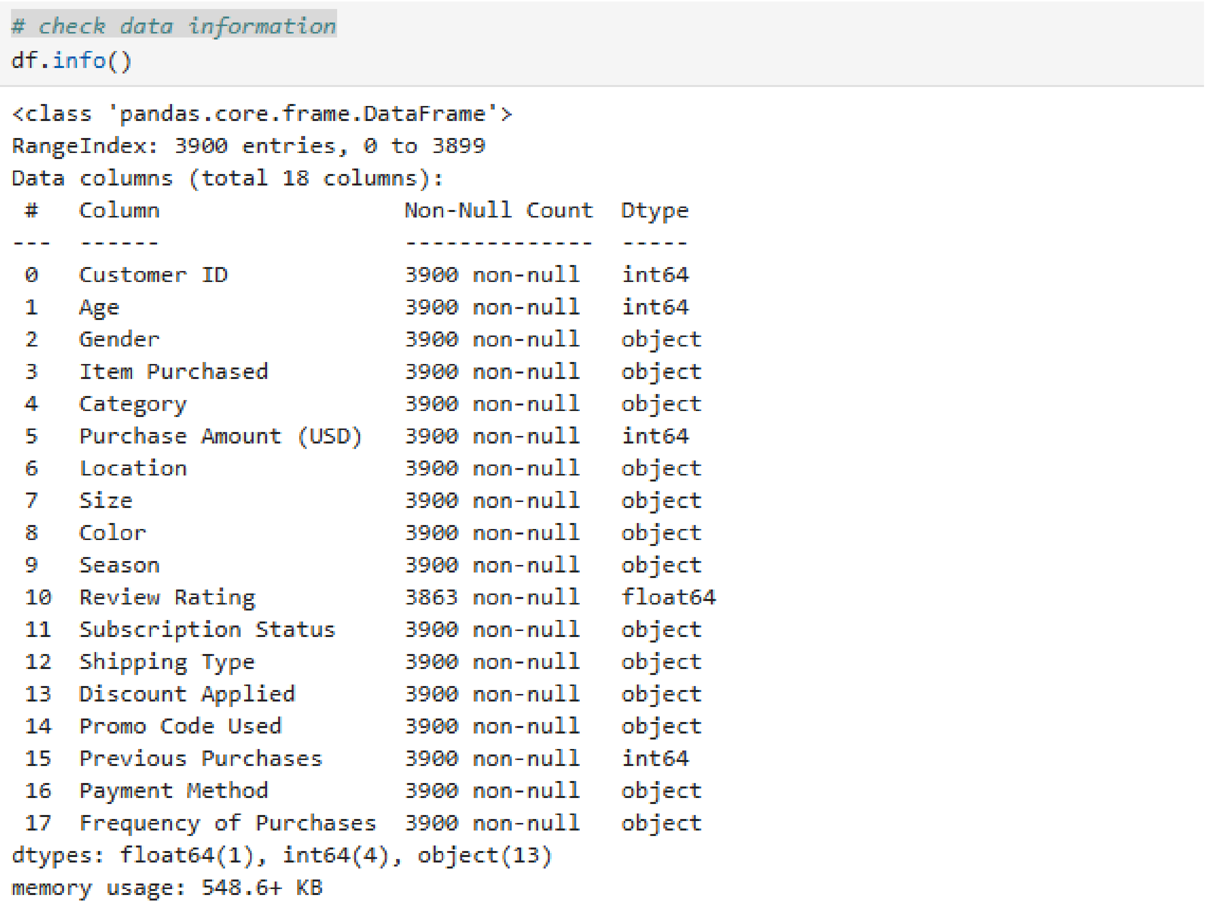
✓ Load the dataset in python



```
# Load the dataset
df = pd.read_csv(r"C:\Users\User\Downloads\customer_shopping_behavior.csv")
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	Paym Met
0	1	55	Male	Blouse	Clothing	53	Kentucky	L	Gray	Winter	3.1	Yes	Express	Yes	Yes	14	Ve
1	2	19	Male	Sweater	Clothing	64	Maine	L	Maroon	Winter	3.1	Yes	Express	Yes	Yes	2	(
2	3	50	Male	Jeans	Clothing	73	Massachusetts	S	Maroon	Spring	3.1	Yes	Free Shipping	Yes	Yes	23	Cr (
3	4	21	Male	Sandals	Footwear	90	Rhode Island	M	Maroon	Spring	3.5	Yes	Next Day Air	Yes	Yes	49	Pa
4	5	45	Male	Blouse	Clothing	49	Oregon	M	Turquoise	Spring	2.7	Yes	Free Shipping	Yes	Yes	31	Pa

✓ Check data information.



```
# check data information
df.info()
```

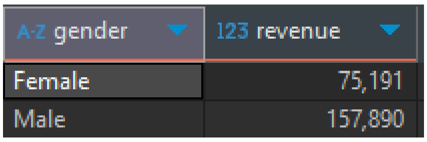
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3900 entries, 0 to 3899
Data columns (total 18 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   Customer ID             3900 non-null   int64  
 1   Age                     3900 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
```

- ✓ Missing Data – Null values were filled with median values for each category so as not to introduce bias in the overall dataset.
- ✓ Column Standardization – Lower case was used for easy use in feature engineering and connection to SQL. ‘Promo code used’ column was also dropped as it had the same information as ‘Discount Applied’ column.
- ✓ Database Integration – Connected the python script to PostgreSQL for data analysis in SQL

4. Data Analysis using SQL

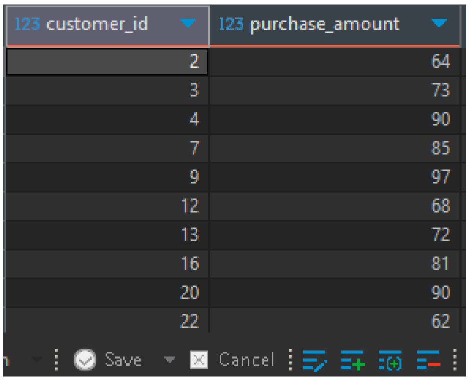
The following analysis was performed to answer key business questions.

```
--Q1. What is the total revenue generated by male vs. female customers?

select gender, SUM(purchase_amount) as revenue
from customer
group by gender
```



A-Z gender	123 revenue
Female	75,191
Male	157,890

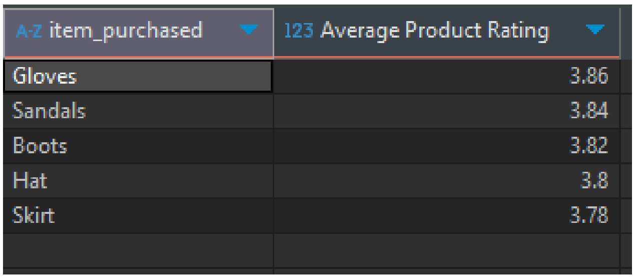
```
--Q2. Which customers used a discount but still spent more than the average
purchase amount?

select customer_id, purchase_amount
from customer
where discount_applied = 'Yes' and purchase_amount >= (select AVG(purchase_amount)
from customer)
```



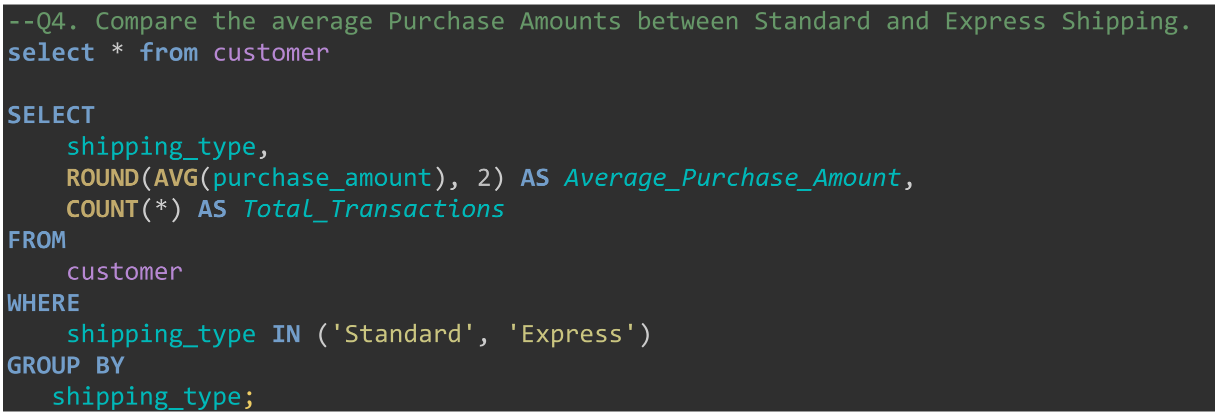
123 customer_id	123 purchase_amount
2	64
3	73
4	90
7	85
9	97
12	68
13	72
16	81
20	90
22	62

```
-- Q3. Which are the top 5 products with the highest average review rating?
```

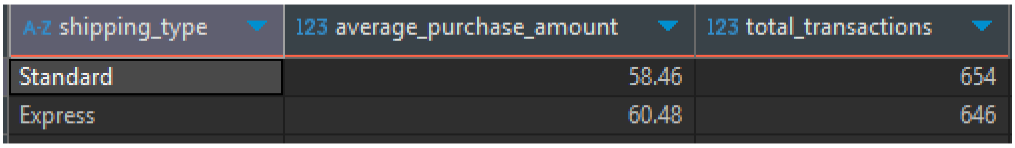
```
select item_purchased, round(avg(review_rating::numeric),2) as "Average Product
Rating"
from customer
group by item_purchased
order by avg(review_rating) desc
limit 5
```



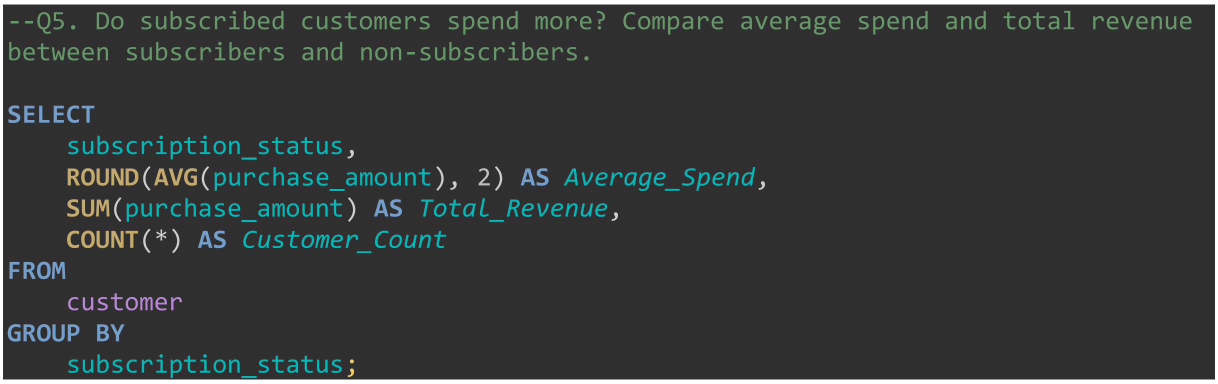
item_purchased	Average Product Rating
Gloves	3.86
Sandals	3.84
Boots	3.82
Hat	3.8
Skirt	3.78



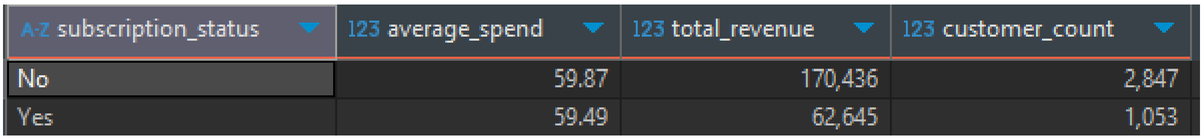
```
--Q4. Compare the average Purchase Amounts between Standard and Express Shipping.
select * from customer

SELECT
    shipping_type,
    ROUND(AVG(purchase_amount), 2) AS Average_Purchase_Amount,
    COUNT(*) AS Total_Transactions
FROM
    customer
WHERE
    shipping_type IN ('Standard', 'Express')
GROUP BY
   shipping_type;
```



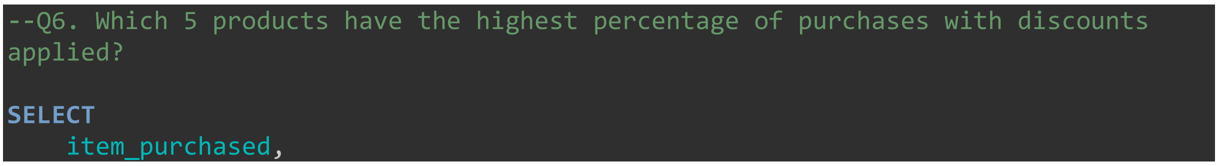
shipping_type	average_purchase_amount	total_transactions
Standard	58.46	654
Express	60.48	646



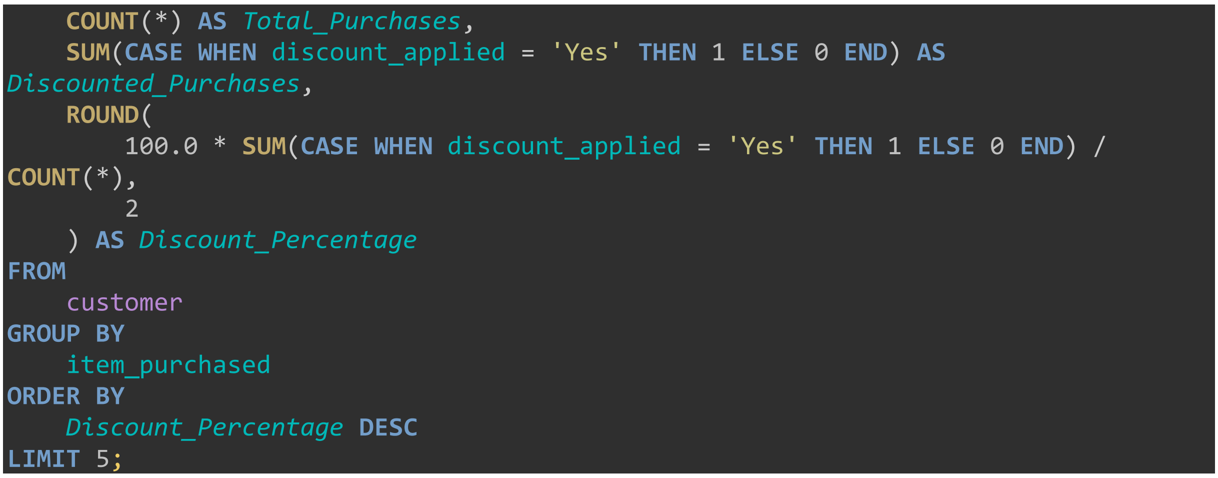
```
--Q5. Do subscribed customers spend more? Compare average spend and total revenue
between subscribers and non-subscribers.

SELECT
    subscription_status,
    ROUND(AVG(purchase_amount), 2) AS Average_Spend,
    SUM(purchase_amount) AS Total_Revenue,
    COUNT(*) AS Customer_Count
FROM
    customer
GROUP BY
    subscription_status;
```



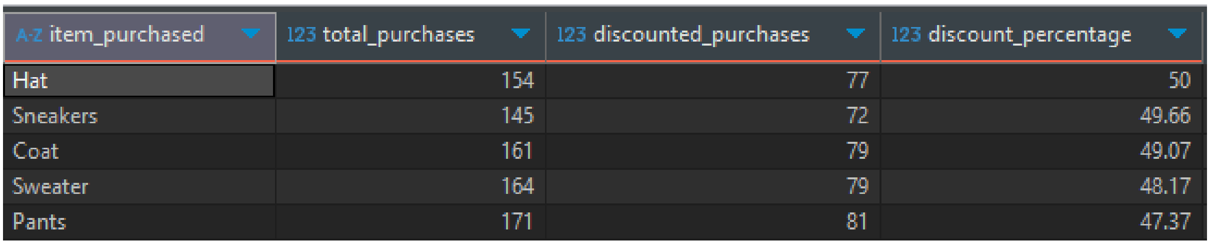
subscription_status	average_spend	total_revenue	customer_count
No	59.87	170,436	2,847
Yes	59.49	62,645	1,053



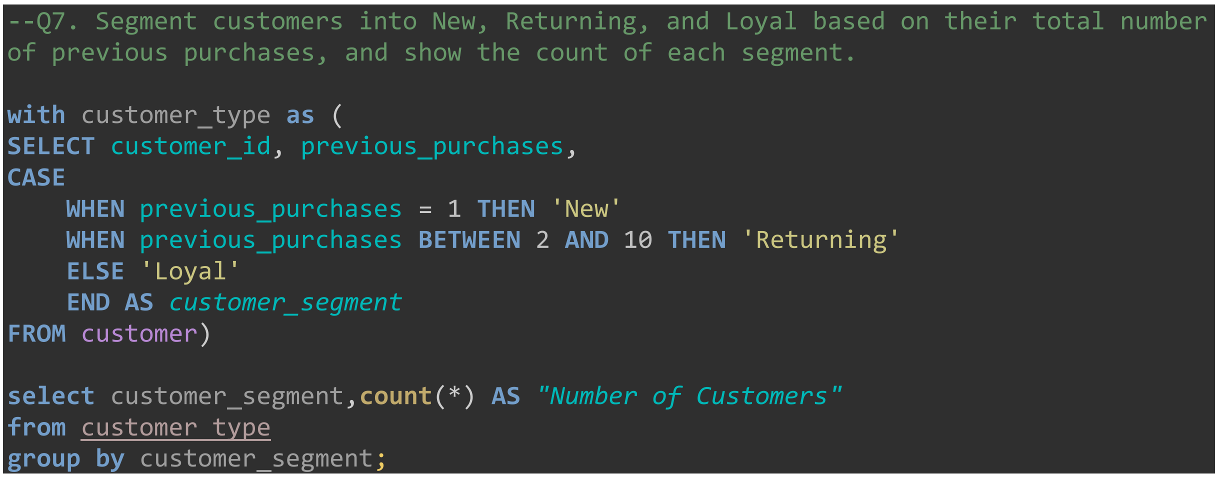
```
--Q6. Which 5 products have the highest percentage of purchases with discounts
applied?

SELECT
    item_purchased,
```



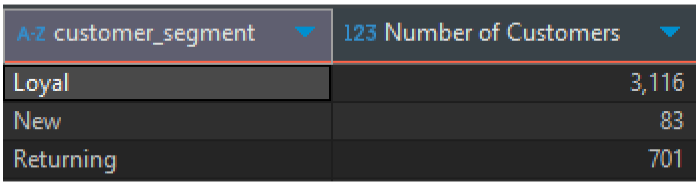
```sql
    COUNT(*) AS Total_Purchases,
    SUM(CASE WHEN discount_applied = 'Yes' THEN 1 ELSE 0 END) AS
Discounted_Purchases,
    ROUND(
        100.0 * SUM(CASE WHEN discount_applied = 'Yes' THEN 1 ELSE 0 END) /
COUNT(*),
        2
    ) AS Discount_Percentage
FROM
    customer
GROUP BY
    item_purchased
ORDER BY
    Discount_Percentage DESC
LIMIT 5;
```



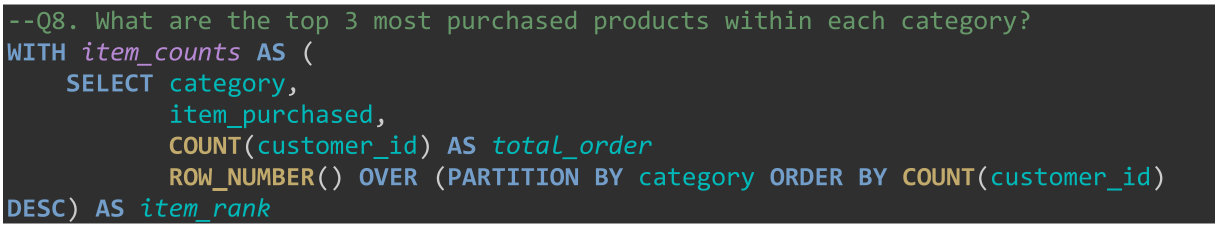
item_purchased	total_purchases	discounted_purchases	discount_percentage
Hat	154	77	50
Sneakers	145	72	49.66
Coat	161	79	49.07
Sweater	164	79	48.17
Pants	171	81	47.37



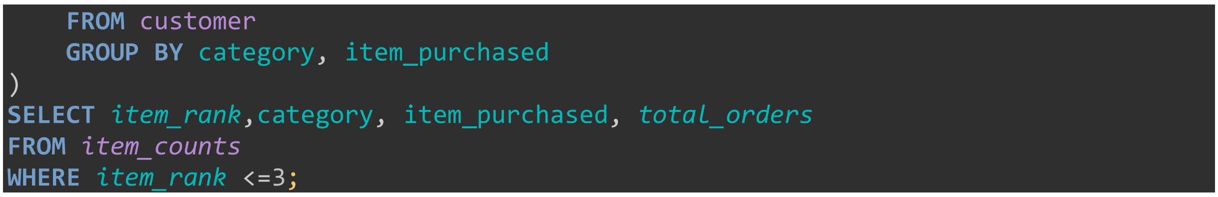
```sql
--Q7. Segment customers into New, Returning, and Loyal based on their total number
of previous purchases, and show the count of each segment.

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)

select customer_segment,count(*) AS "Number of Customers"
from customer_type
group by customer_segment;
```



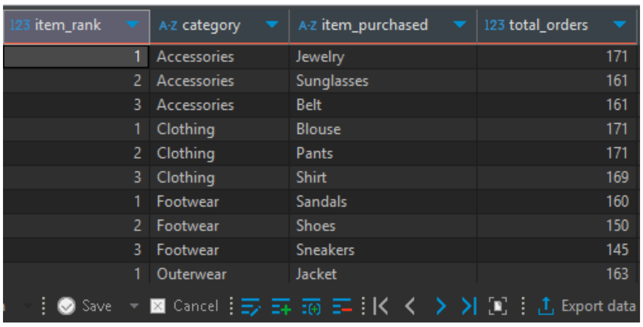
customer_segment	Number of Customers
Loyal	3,116
New	83
Returning	701



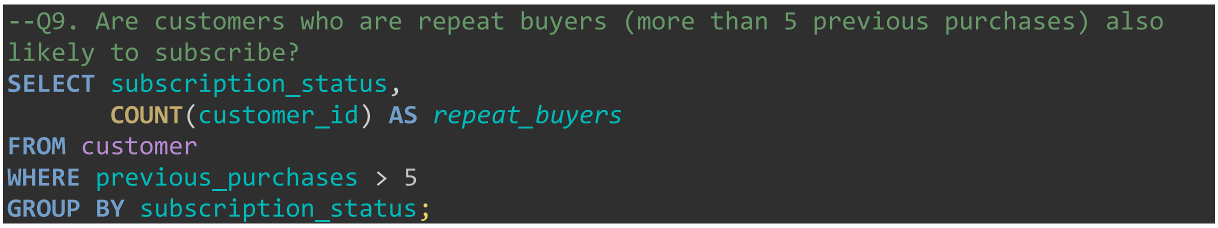
```sql
--Q8. What are the top 3 most purchased products within each category?
WITH item_counts AS (
    SELECT category,
           item_purchased,
           COUNT(customer_id) AS total_order
           ROW_NUMBER() OVER (PARTITION BY category ORDER BY COUNT(customer_id)
DESC) AS item_rank
```



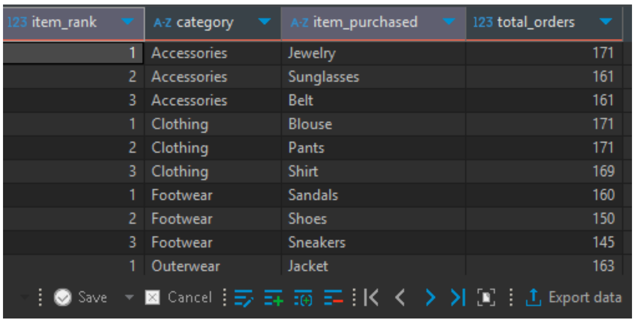
```
    FROM customer
    GROUP BY category, item_purchased
)
SELECT item_rank,category, item_purchased, total_orders
FROM item_counts
WHERE item_rank <=3;
```



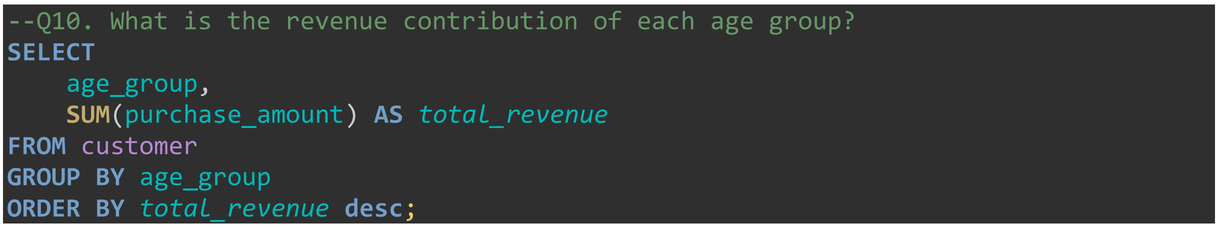
item_rank	category	item_purchased	total_orders
1	Accessories	Jewelry	171
2	Accessories	Sunglasses	161
3	Accessories	Belt	161
1	Clothing	Blouse	171
2	Clothing	Pants	171
3	Clothing	Shirt	169
1	Footwear	Sandals	160
2	Footwear	Shoes	150
3	Footwear	Sneakers	145
1	Outerwear	Jacket	163



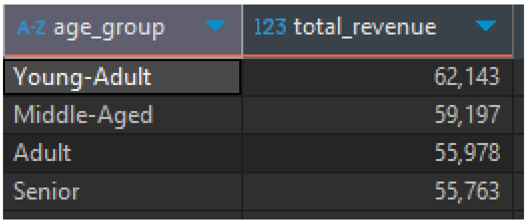
```
--Q9. Are customers who are repeat buyers (more than 5 previous purchases) also
likely to subscribe?
SELECT subscription_status,
       COUNT(customer_id) AS repeat_buyers
FROM customer
WHERE previous_purchases > 5
GROUP BY subscription_status;
```



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

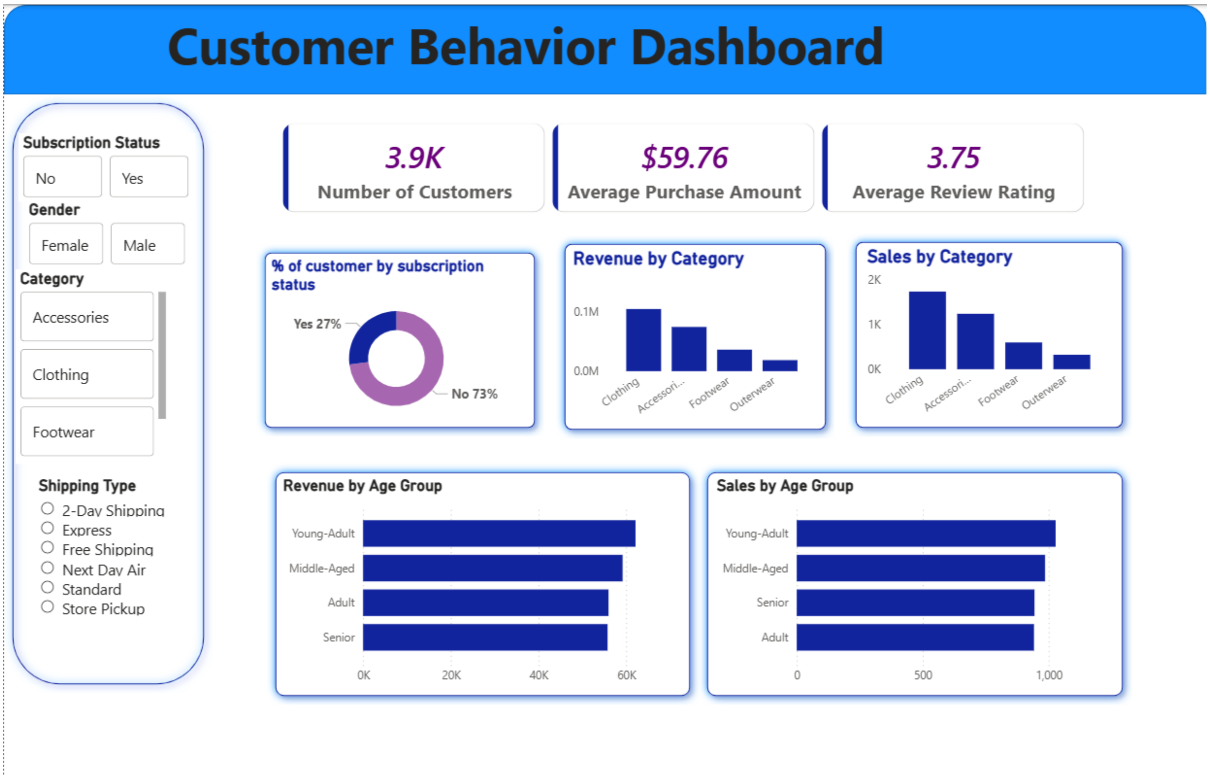


```
--Q10. What is the revenue contribution of each age group?
SELECT
    age_group,
    SUM(purchase_amount) AS total_revenue
FROM customer
GROUP BY age_group
ORDER BY total_revenue desc;
```



age_group	total_revenue
Young-Adult	62,143
Middle-Aged	59,197
Adult	55,978
Senior	55,763

PowerBI Dashboard



5. Business Recommendations

1. Pivot Subscription Strategy from **Spending** to **Frequency**

The analysis revealed that there is no statistically significant difference in the average purchase amount between subscribers ($59.49) and non-subscribers ($59.87). This indicates that the current subscription model does not encourage customers to buy more expensive items. Instead of focusing on transaction value, you should pivot the subscription value proposition toward increasing purchase frequency.

2. Optimize Inventory for **Clothing** and **Accessories**.

The data shows these categories significantly outperform Footwear and Outerwear. By reallocating budget from lower-performing categories like Outerwear (which has a more limited seasonal peak) into high-velocity Accessories, you can improve stock turnover rates and reduce the capital tied up in slow-moving inventory.

3. Implement a "High-Value Discount" Retention Program

Our analysis identified a segment of **839 customers** who used a discount but still spent more than the average purchase amount ($59.76). These are your most valuable "promotion-sensitive" customers. Rather than offering broad discounts to everyone, you should create a targeted retention program for this specific group. Offering them "threshold-based" discounts (e.g., "Spend $100, Get $20 Off") will likely yield higher returns than flat percentage discounts, as this group has already demonstrated a willingness to maintain high transaction values even when seeking deals.