Customer Shopping Behavior Analysis Report

Project Title: Customer Shopping Behavior Analysis

Table of contents

1. Business Problem Statement

1.1 Business Context

A leading retail company is experiencing shifts in purchasing patterns across different customer demographics and product categories. While sales data is available, the management team lacks deep visibility into what specifically drives consumer decisions—such as the impact of discounts, seasonal trends, and review ratings. To maintain a competitive edge, the company needs to transition from reactive sales tracking to proactive, data-driven strategy planning.

1.2 Problem Statement

The company currently struggles to identify which distinct factors (e.g., age groups, shipping preferences, payment methods) contribute most to customer loyalty and repeat purchases. Without this insight, marketing campaigns are generic, discount strategies may be wasteful, and high-potential customer segments remain untapped, leading to missed revenue opportunities and lower customer satisfaction.

1.3 Project Objective

The goal of this analysis is to leverage the "Customer Shopping Trends" dataset to uncover actionable patterns in consumer behavior. By connecting data analysis (Python/SQL) with visual storytelling (Power BI), this project aims to provide the management team with evidence-based recommendations to optimize product offerings, refine marketing strategies, and increase long-term customer retention.

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. Deliverables

1. Data Preparation & Modeling (Python): Clean and transform the raw dataset for analysis.

2. Data Analysis (SQL): Organize the data into a structured format, simulate business transactions, and run queries to extract insights on customer segments, loyalty, and purchase drivers.

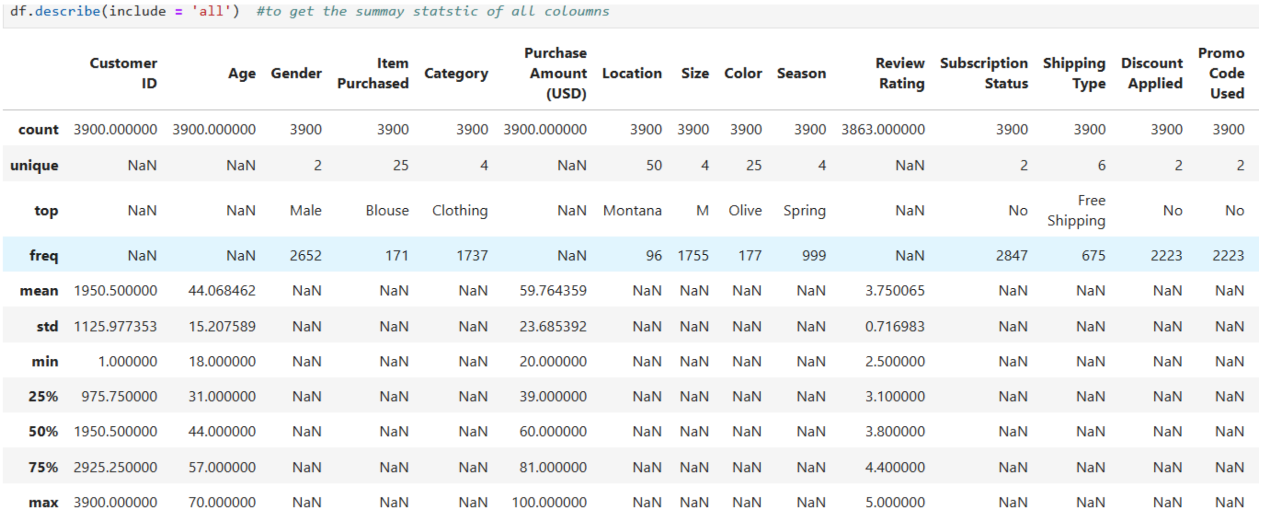
3. Visualization & Insights (Power BI): Build an interactive dashboard that highlights key patterns and trends, enabling stakeholders to make data-driven decisions.

4. GitHub Repository: Include all Python scripts, SQL queries, and dashboard files in a well-structured repository.

4. Exploratory Data Analysis using Python

Data Loading: Imported the dataset using pandas.

Initial Exploration: Used df.info() to check structure and .describe() for summary statistics.



```
df.describe(include = 'all')  #to get the summay statstic of all coloumns
```

	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
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 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:
- I. Created age_group column by binning customer ages.
- II. Created purchase_frequency_days column from purchase data.

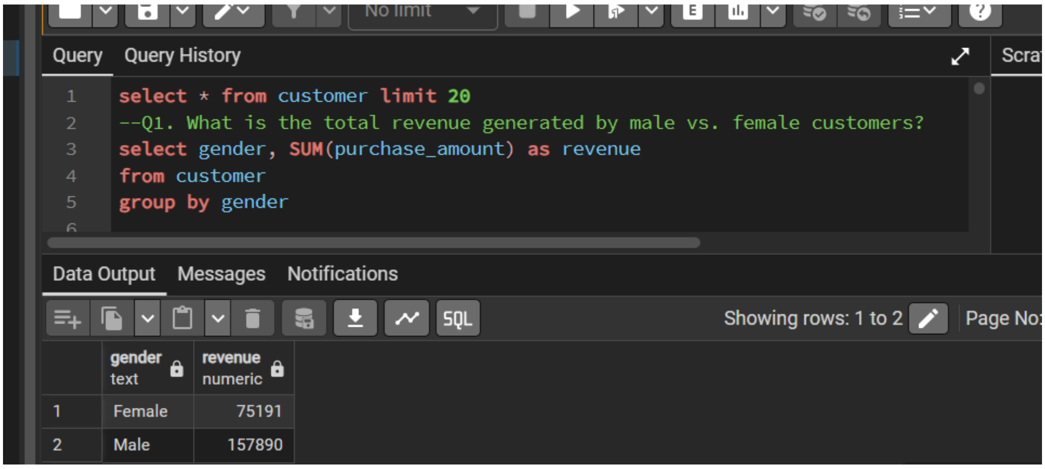
Data Consistency Check: Verified if discount_applied and promo_code_used were redundant; dropped promo_code_used.

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

5. Data Analysis using SQL (Business Transactions)

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

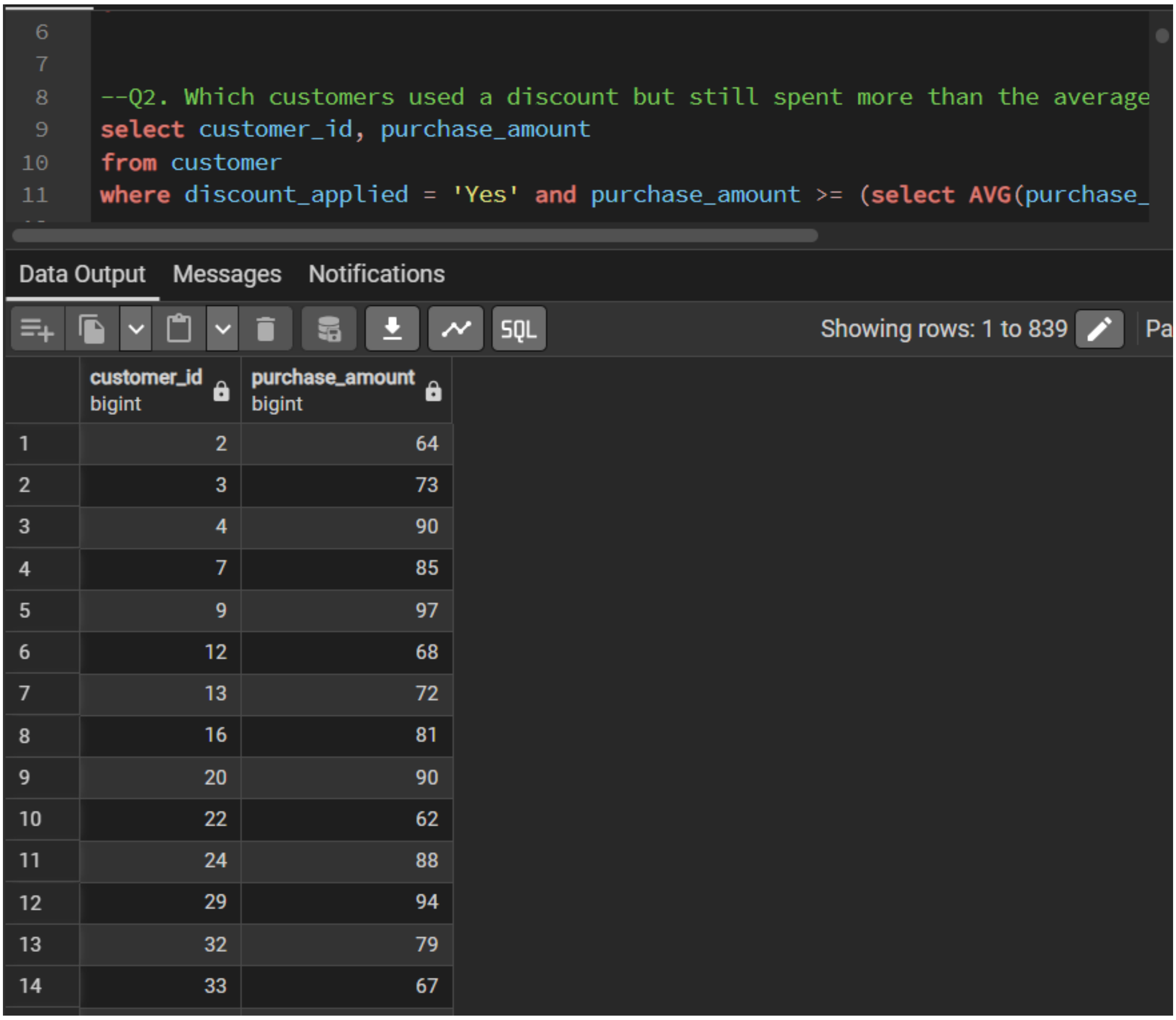
1. Revenue by Gender – Compared total revenue generated by male vs. female customers.



```
select * from customer limit 20
--Q1. What is the total revenue generated by male vs. female customers?
select gender, SUM(purchase_amount) as revenue
from customer
group by gender
```

	gender text	revenue numeric
1	Female	75191
2	Male	157890

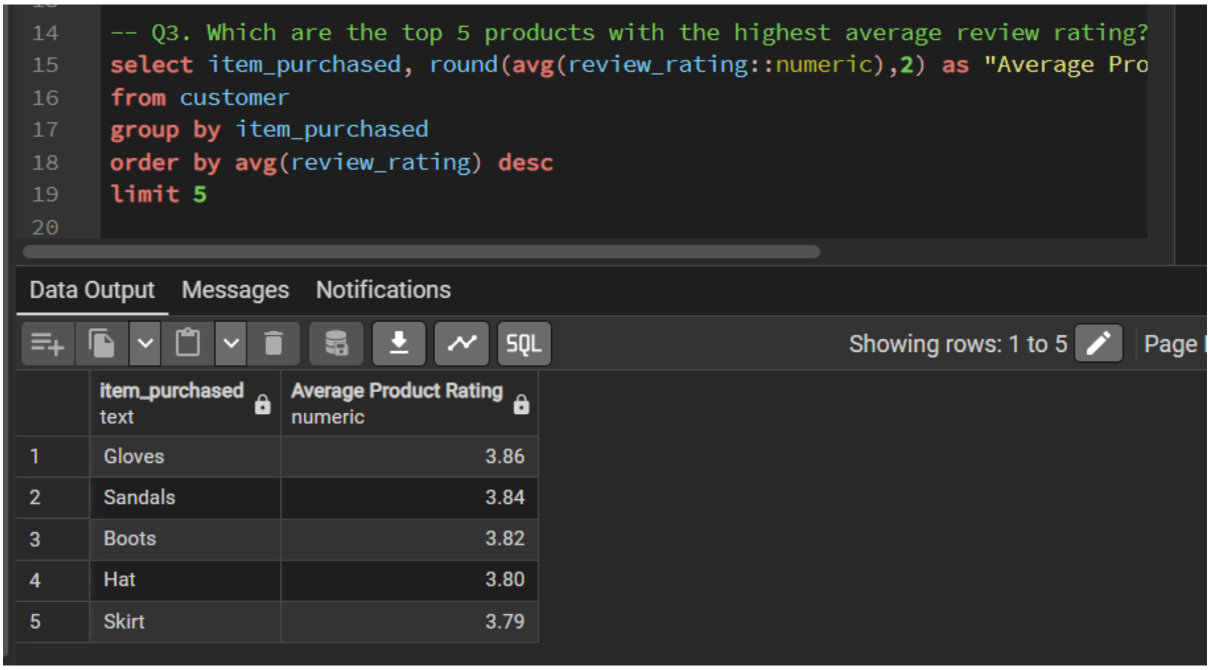
2. High-Spending Discount Users – Identified customers who used discounts but still spent above the average purchase amount.



```
--Q2. Which customers used a discount but still spent more than the average
select customer_id, purchase_amount
from customer
where discount_applied = 'Yes' and purchase_amount >= (select AVG(purchase_
```

	customer_id bigint	purchase_amount bigint
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

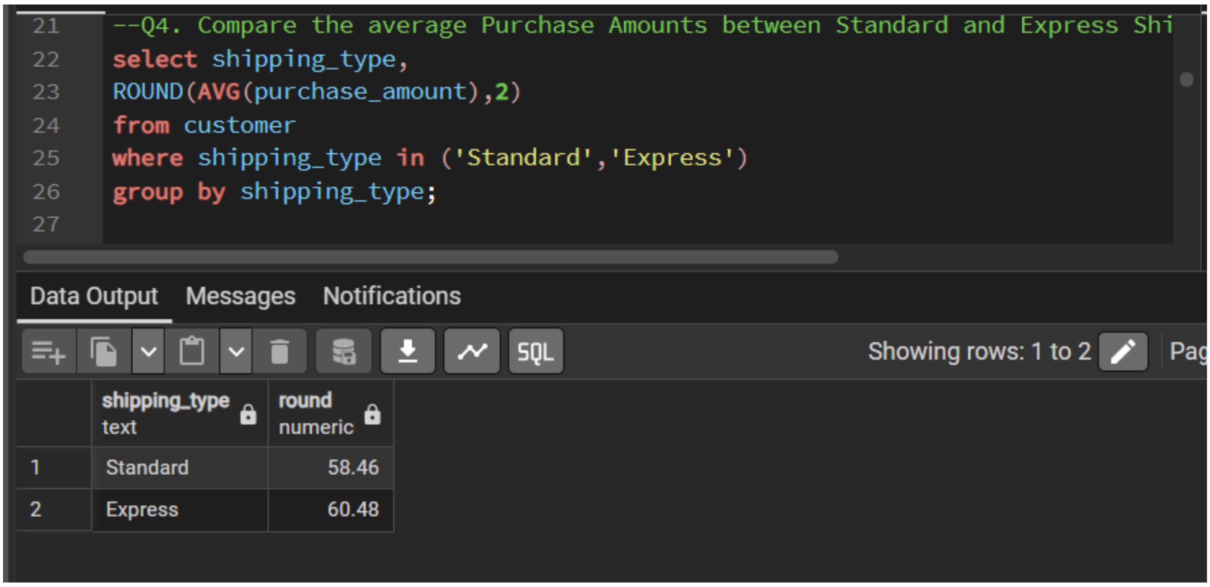
3. Top 5 Products by Rating – Found products with the highest average review ratings.



```
-- Q3. Which are the top 5 products with the highest average review rating?
select item_purchased, round(avg(review_rating::numeric),2) as "Average Pro
from customer
group by item_purchased
order by avg(review_rating) desc
limit 5
```

	item_purchased text	Average Product Rating numeric
1	Gloves	3.86
2	Sandals	3.84
3	Boots	3.82
4	Hat	3.80
5	Skirt	3.79

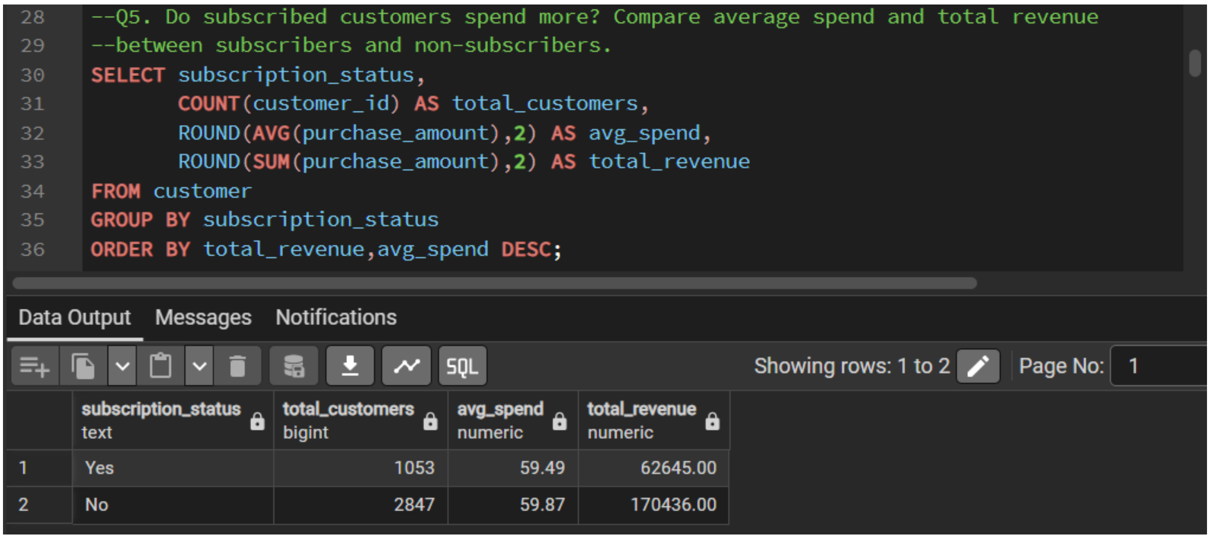
4. Shipping Type Comparison – Compared average purchase amounts between Standard and Express shipping



```
--Q4. Compare the average Purchase Amounts between Standard and Express Shi
select shipping_type,
ROUND(AVG(purchase_amount),2)
from customer
where shipping_type in ('Standard','Express')
group by shipping_type;
```

	shipping_type text	round numeric
1	Standard	58.46
2	Express	60.48

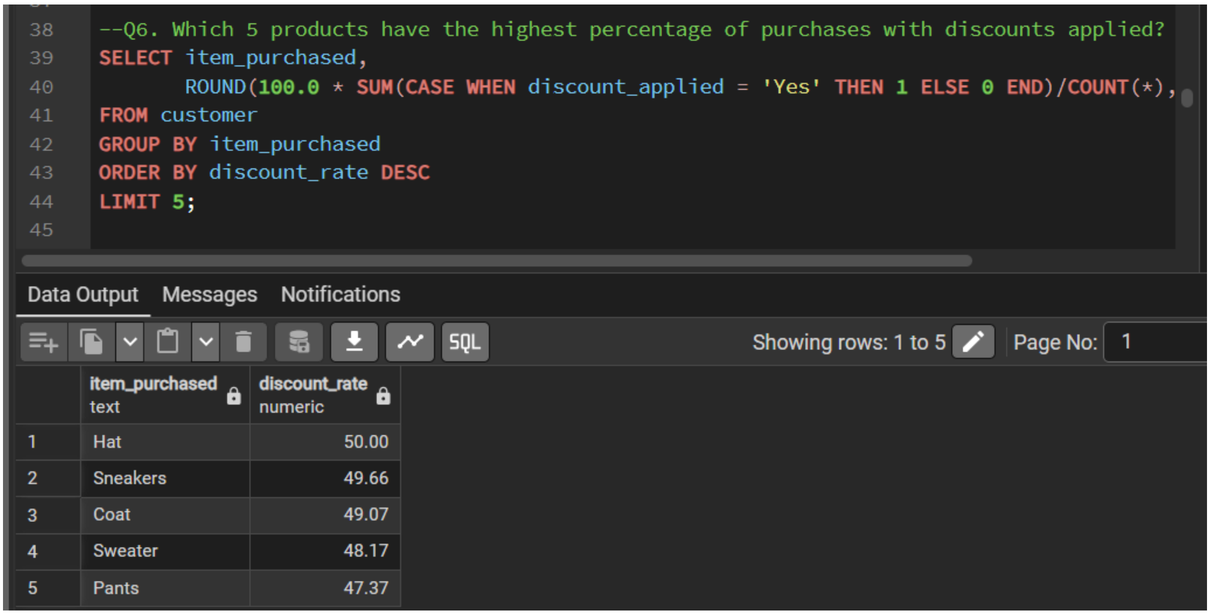
5. Subscribers vs. Non-Subscribers – Compared average spend and total revenue across subscription status



```
--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_customers,
       ROUND(AVG(purchase_amount),2) AS avg_spend,
       ROUND(SUM(purchase_amount),2) AS total_revenue
FROM customer
GROUP BY subscription_status
ORDER BY total_revenue,avg_spend DESC;
```

	subscription_status text	total_customers bigint	avg_spend numeric	total_revenue numeric
1	Yes	1053	59.49	62645.00
2	No	2847	59.87	170436.00

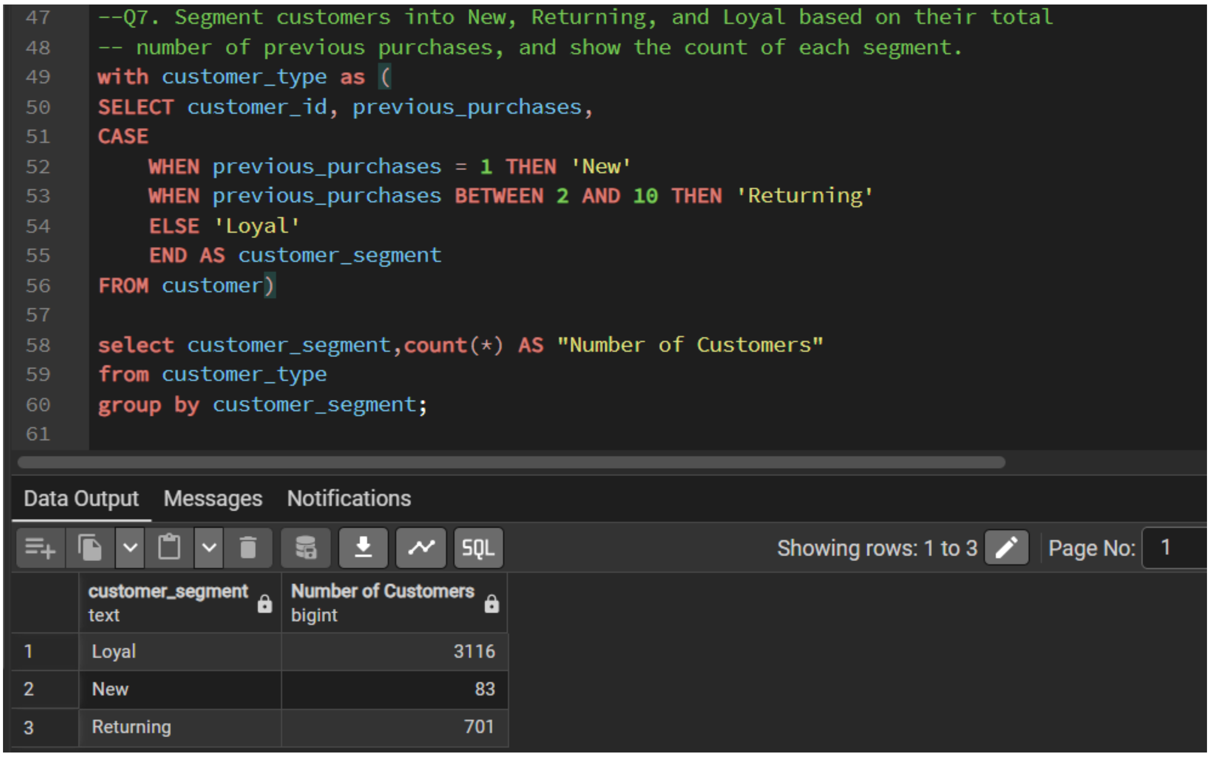
6. Discount-Dependent Products – Identified 5 products with the highest percentage of discounted purchases



```
--Q6. Which 5 products have the highest percentage of purchases with discounts applied?
SELECT item_purchased,
       ROUND(100.0 * SUM(CASE WHEN discount_applied = 'Yes' THEN 1 ELSE 0 END)/COUNT(*),
FROM customer
GROUP BY item_purchased
ORDER BY discount_rate DESC
LIMIT 5;
```

	item_purchased text	discount_rate numeric
1	Hat	50.00
2	Sneakers	49.66
3	Coat	49.07
4	Sweater	48.17
5	Pants	47.37

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

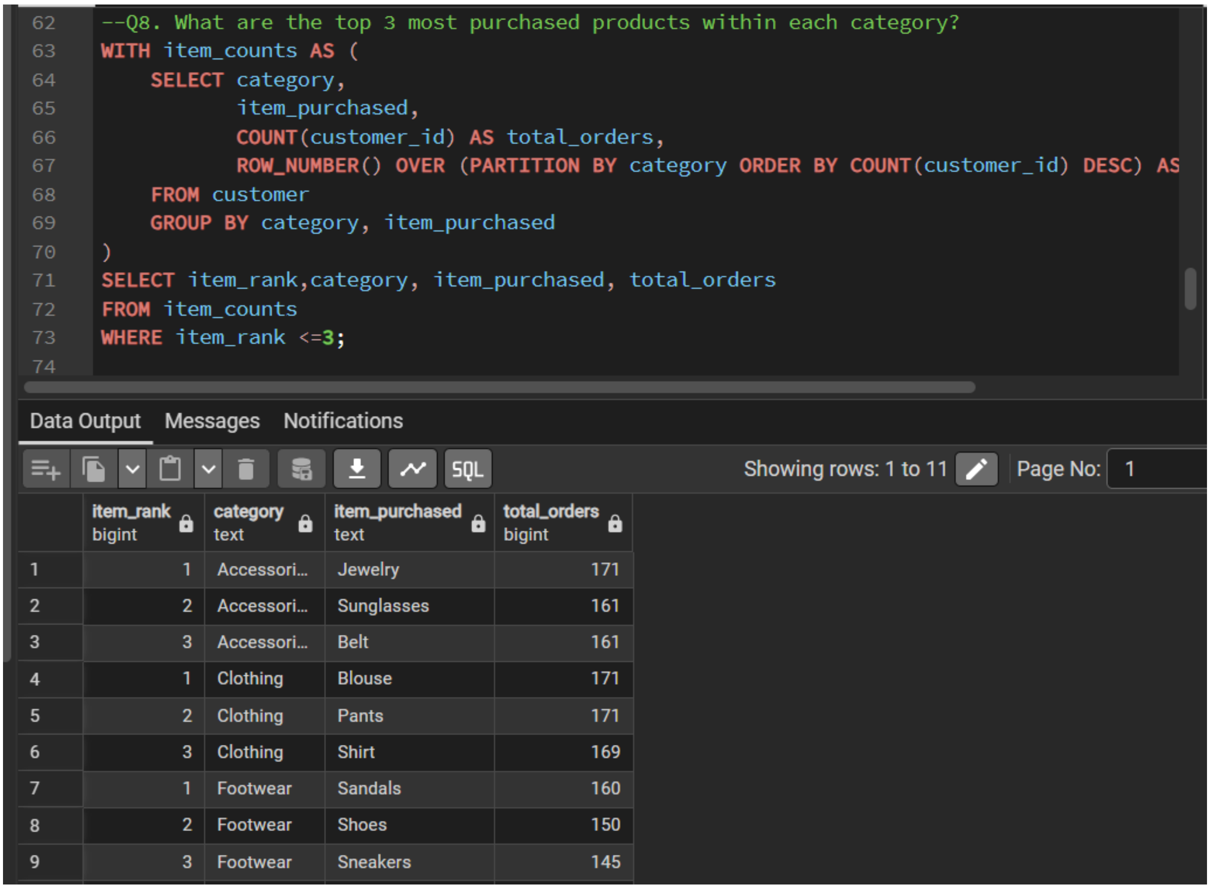


```sql
--Q7. Segment customers into New, Returning, and Loyal based on their total
-- number of previous purchases, and show the count of each segment.
with customer_type as (
SELECT customer_id, previous_purchases,
CASE
    WHEN previous_purchases = 1 THEN 'New'
    WHEN previous_purchases BETWEEN 2 AND 10 THEN 'Returning'
    ELSE 'Loyal'
    END AS customer_segment
FROM customer)

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

	customer_segment text	Number of Customers bigint
1	Loyal	3116
2	New	83
3	Returning	701

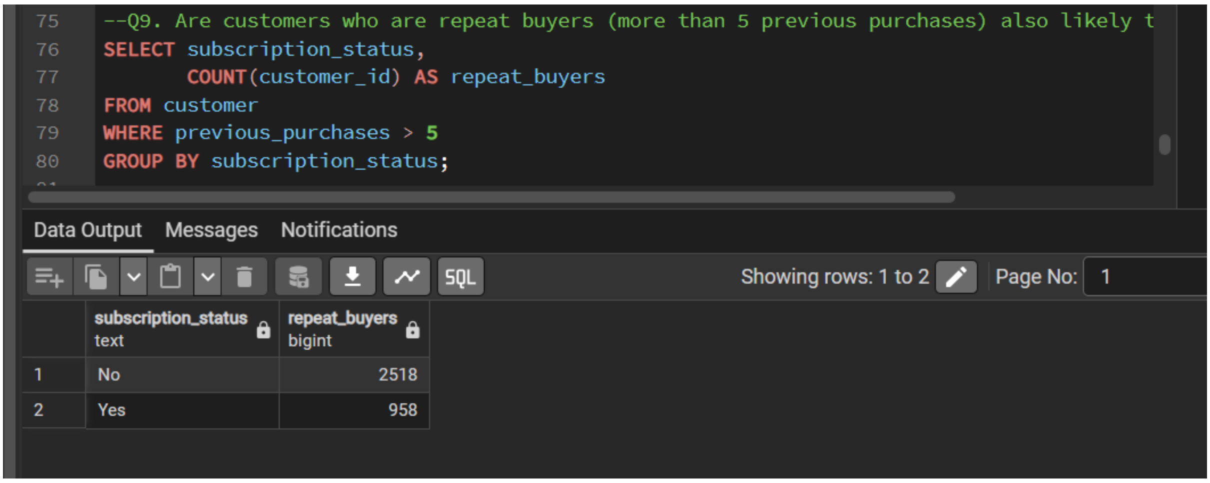
8. Top 3 Products per Category – Listed the most purchased products within each category



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

	item_rank bigint	category text	item_purchased text	total_orders bigint
1	1	Accessori...	Jewelry	171
2	2	Accessori...	Sunglasses	161
3	3	Accessori...	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

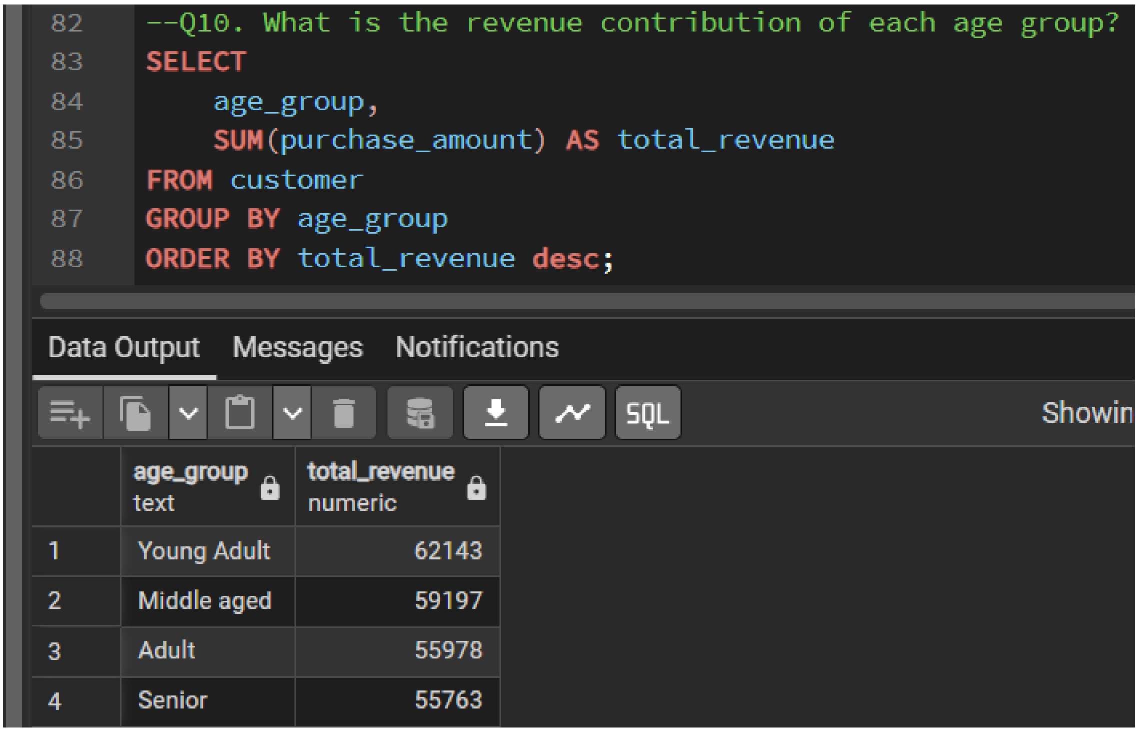
9. Repeat Buyers & Subscriptions – Checked whether customers with >5 purchases are more likely to subscribe



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

	subscription_status text	repeat_buyers bigint
1	No	2518
2	Yes	958

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

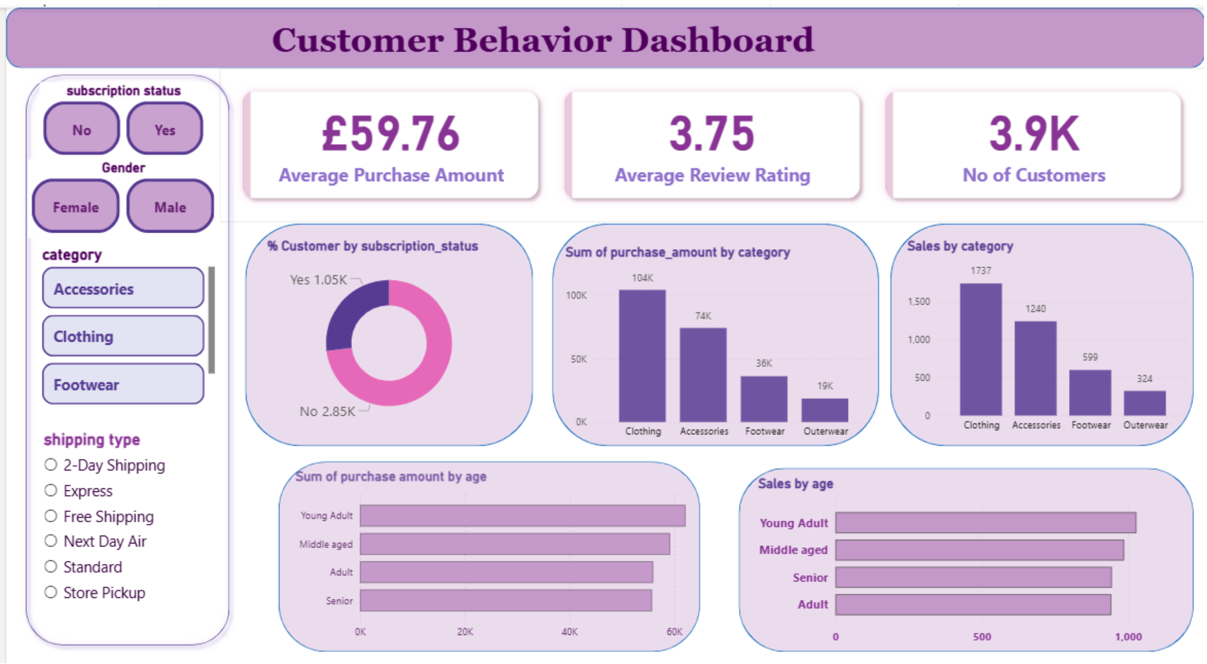


```sql
--Q10. What is the revenue contribution of each age group?
SELECT
    age_group,
    SUM(purchase_amount) AS total_revenue
FROM customer
GROUP BY age_group
ORDER BY total_revenue desc;
```

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

6. Dashboard in Power BI

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



7.Conclusion

This project successfully transformed raw transaction data into actionable business intelligence. By integrating Python for data quality, SQL for deep segmentation, and Power BI for visual storytelling, we moved beyond simple reporting to uncover the *drivers* of customer behavior.The analysis highlighted that the company's "sweet spot" lies with Adult customers (30-50) and the Clothing category.