CUSTOMER INSIGHTS ANALYSIS USING SQL – PORTFOLIO PROJECT.

INTRODUCTION.

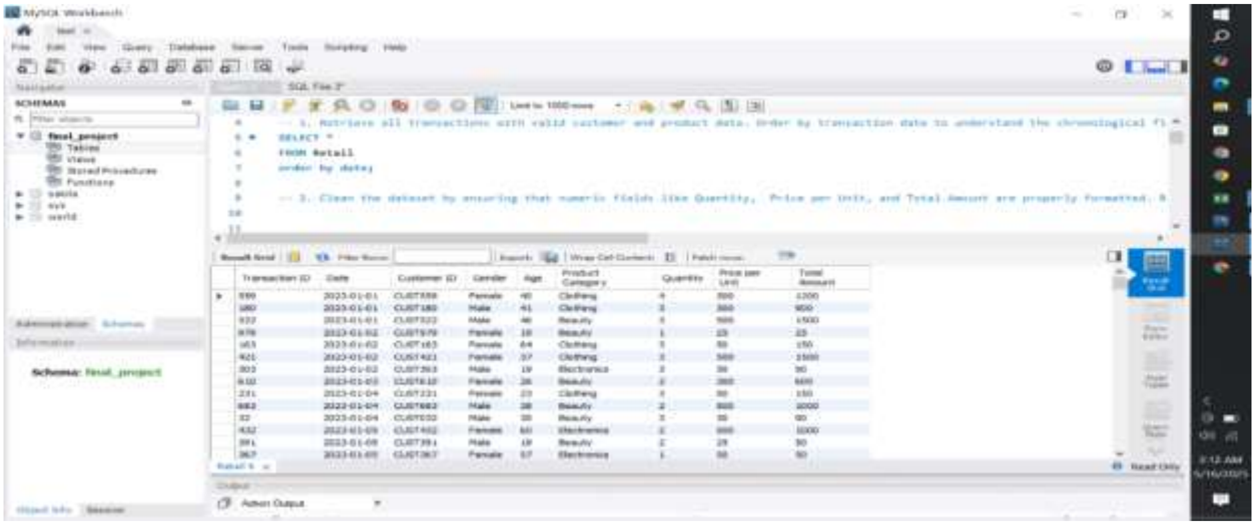
Understanding customer behavior is at the heart of modern retail strategy. This project, **“Customer Insights Analysis Using SQL – Portfolio Project,”** explores a retail dataset using Structured Query Language (SQL) to uncover meaningful insights about customer purchasing patterns, product performance, and sales trends.
The goal of this analysis is to turns raw transaction data into insights to guide strategy in marketing, segmentation, and product performance.

1. Retrieve all transactions with valid customer and product data. Order by transaction date to understand the chronological flow of purchases.

SELECT *

FROM Retail

order by date;



```
-- 1. Retrieve all transactions with valid customer and product data. Order by transaction date to understand the chronological fl
SELECT *
FROM Retail
order by date;

-- 2. Clean the dataset by ensuring that numeric fields like Quantity,  Price per Unit, and Total Amount are properly formatted. R
```

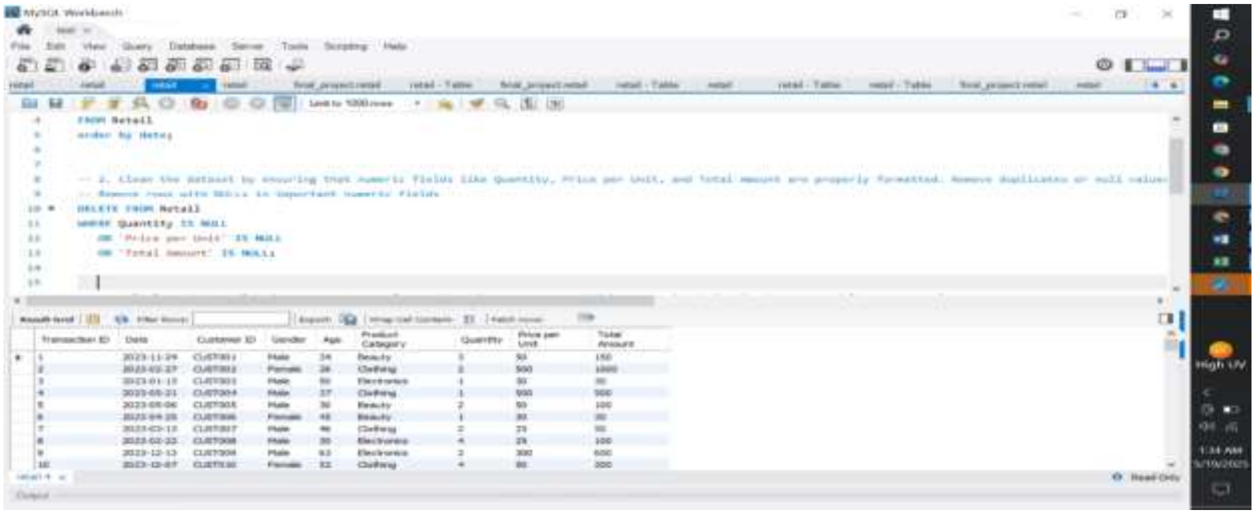
Transaction ID	Date	Customer ID	Gender	Age	Product Category	Quantity	Price per Unit	Total Amount
559	2023-01-01	CUST559	Female	40	Clothing	4	300	1200
180	2023-01-01	CUST180	Male	41	Clothing	3	300	900
522	2023-01-01	CUST522	Male	46	Beauty	3	500	1500
679	2023-01-02	CUST679	Female	18	Beauty	1	25	25
163	2023-01-02	CUST163	Female	64	Clothing	3	50	150
421	2023-01-02	CUST421	Female	37	Clothing	3	500	1500
393	2023-01-02	CUST393	Male	19	Electronics	3	30	90
610	2023-01-03	CUST610	Female	26	Beauty	2	300	600
231	2023-01-04	CUST231	Female	23	Clothing	3	50	150
683	2023-01-04	CUST683	Male	38	Beauty	2	500	1000
32	2023-01-04	CUST032	Male	30	Beauty	3	30	90
432	2023-01-05	CUST432	Female	60	Electronics	2	500	1000
391	2023-01-05	CUST391	Male	19	Beauty	2	25	50
367	2023-01-05	CUST367	Female	57	Electronics	1	50	50

➢ Objective

The screenshot shows a list of retail transactions sorted by date, including customer details, product types, and total amounts. It helps analyze purchase patterns over time.

2. Clean the dataset by ensuring that numeric fields like Quantity, Price per Unit, and Total Amount are properly formatted. Remove duplicates or null values if any exist. Remove rows with NULLs in important numeric fields

```
DELETE FROM Retail

WHERE Quantity IS NULL

   OR `Price per Unit` IS NULL

   OR `Total Amount` IS NULL;
```



```
FROM Retail
order by date;


-- 2. Clean the dataset by ensuring that numeric fields like Quantity, Price per Unit, and Total Amount are properly formatted. Remove duplicates or null values
-- Remove rows with NULLs in important numeric fields
DELETE FROM Retail
WHERE Quantity IS NULL
   OR `Price per Unit` IS NULL
   OR `Total Amount` IS NULL;
```

Transaction ID	Date	Customer ID	Gender	Age	Product Category	Quantity	Price per Unit	Total Amount
1	2023-11-24	CUST001	Male	34	Beauty	3	50	150
2	2023-02-27	CUST002	Female	26	Clothing	2	500	1000
3	2023-01-13	CUST003	Male	50	Electronics	1	30	30
4	2023-05-21	CUST004	Male	37	Clothing	1	500	500
5	2023-05-06	CUST005	Male	30	Beauty	2	50	100
6	2023-04-25	CUST006	Female	45	Beauty	1	30	30
7	2023-03-13	CUST007	Male	46	Clothing	2	25	50
8	2023-02-22	CUST008	Male	30	Electronics	4	25	100
9	2023-12-13	CUST009	Male	63	Electronics	2	300	600
10	2023-10-07	CUST010	Female	52	Clothing	4	50	200

➢ Objective

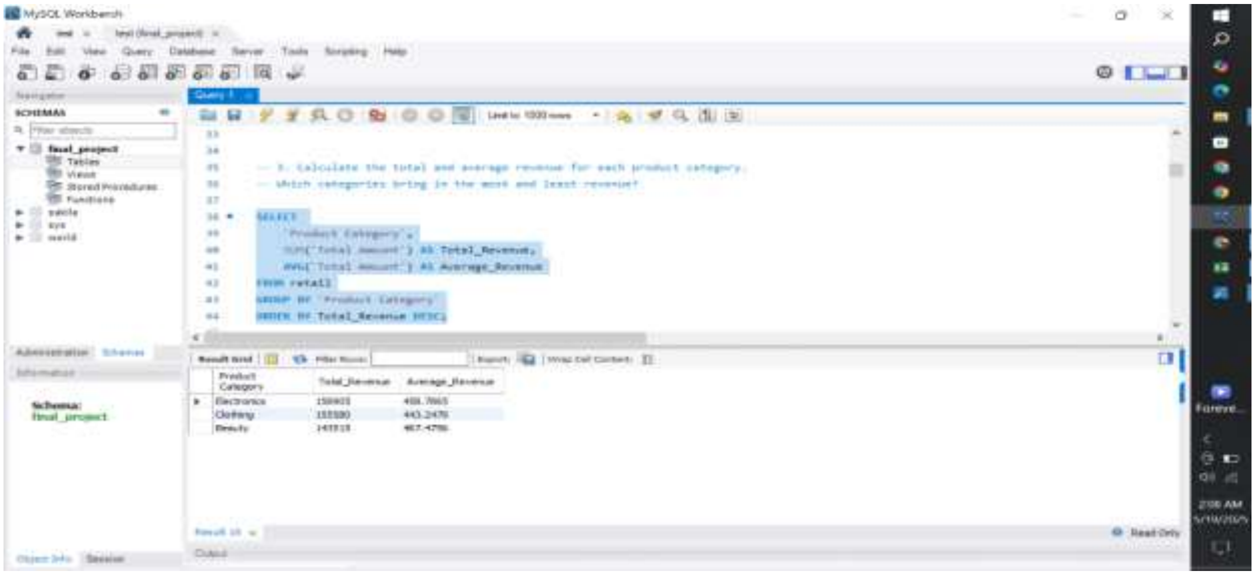
Ensure data integrity by removing nulls and formatting numeric fields properly for accurate analysis.

3. Calculate the total and average revenue for each product category.

Which categories bring in the most and least revenue?

```
SELECT

   `Product Category`,

   SUM(`Total Amount`) AS Total Revenue,

   AVG(`Total Amount`) AS Average Revenue

FROM retail

GROUP BY `Product Category`
```

ORDER BY Total Revenue DESC;



```
-- 3. Calculate the total and average revenue for each product category.
-- Which categories bring in the most and least revenue?

SELECT
    `Product Category`,
    SUM(`Total Amount`) AS Total_Revenue,
    AVG(`Total Amount`) AS Average_Revenue
FROM retail
GROUP BY `Product Category`
ORDER BY Total_Revenue DESC;
```

Product Category	Total_Revenue	Average_Revenue
Electronics	156905	458.7865
Clothing	155580	443.2479
Beauty	143515	467.4756

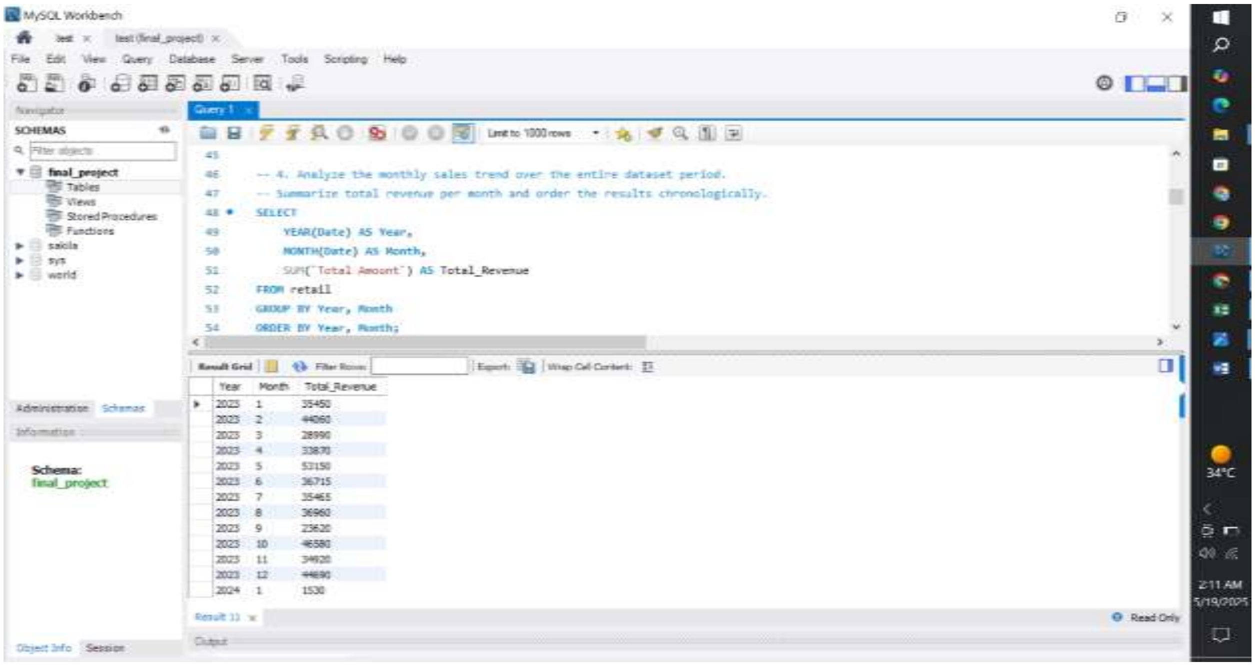
➢ Objective

Determine which product categories generate the most and least income.

4. Analyze the monthly sales trend over the entire dataset period. Summarize total revenue per month and order the results chronologically.

```
SELECT
    YEAR(Date) AS Year,
    MONTH(Date) AS Month,
    SUM(`Total Amount`) AS Total Revenue
FROM retail
GROUP BY Year, Month
```

ORDER BY Year, Month;



```
-- 4. Analyze the monthly sales trend over the entire dataset period.
-- Summarize total revenue per month and order the results chronologically.
SELECT
    YEAR(Date) AS Year,
    MONTH(Date) AS Month,
    SUM(`Total Amount`) AS Total_Revenue
FROM retail
GROUP BY Year, Month
ORDER BY Year, Month;
```

Year	Month	Total_Revenue
2023	1	35450
2023	2	44060
2023	3	28990
2023	4	33870
2023	5	53150
2023	6	36715
2023	7	35465
2023	8	36960
2023	9	23620
2023	10	46580
2023	11	34920
2023	12	44690
2024	1	1530

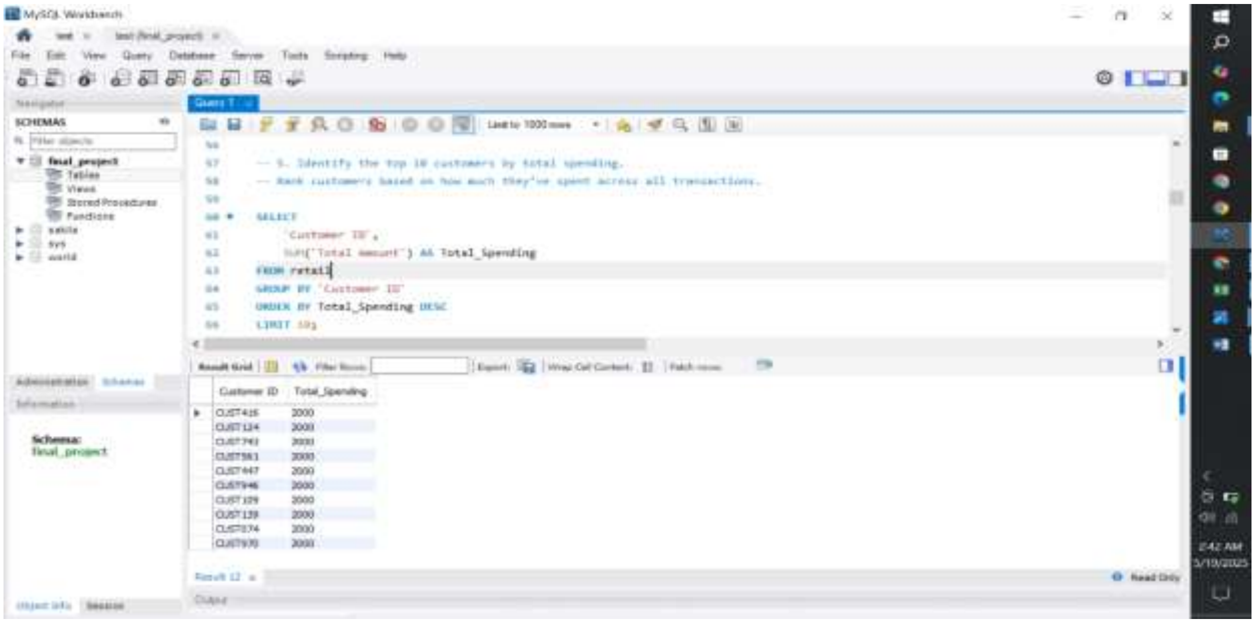
➢ Objective

Monitor revenue performance across months to identify seasonal trends or changes in business growth.

5. Identify the top 10 customers by total spending. Rank customers based on how much they've spent across all transactions.

```
SELECT
  `Customer ID`,
  SUM(`Total Amount`) AS Total Spending
FROM retail
GROUP BY `Customer ID`
ORDER BY Total Spending DESC
```

```
LIMIT 10;
```



```
-- 5. Identify the top 10 customers by total spending.
-- Rank customers based on how much they've spent across all transactions.

SELECT
    `Customer ID`,
    SUM(`Total Amount`) AS Total_Spending
FROM retail
GROUP BY `Customer ID`
ORDER BY Total_Spending DESC
LIMIT 10;
```

Customer ID	Total_Spending
CUST416	2000
CUST124	2000
CUST743	2000
CUST561	2000
CUST447	2000
CUST946	2000
CUST109	2000
CUST139	2000
CUST074	2000
CUST970	2000

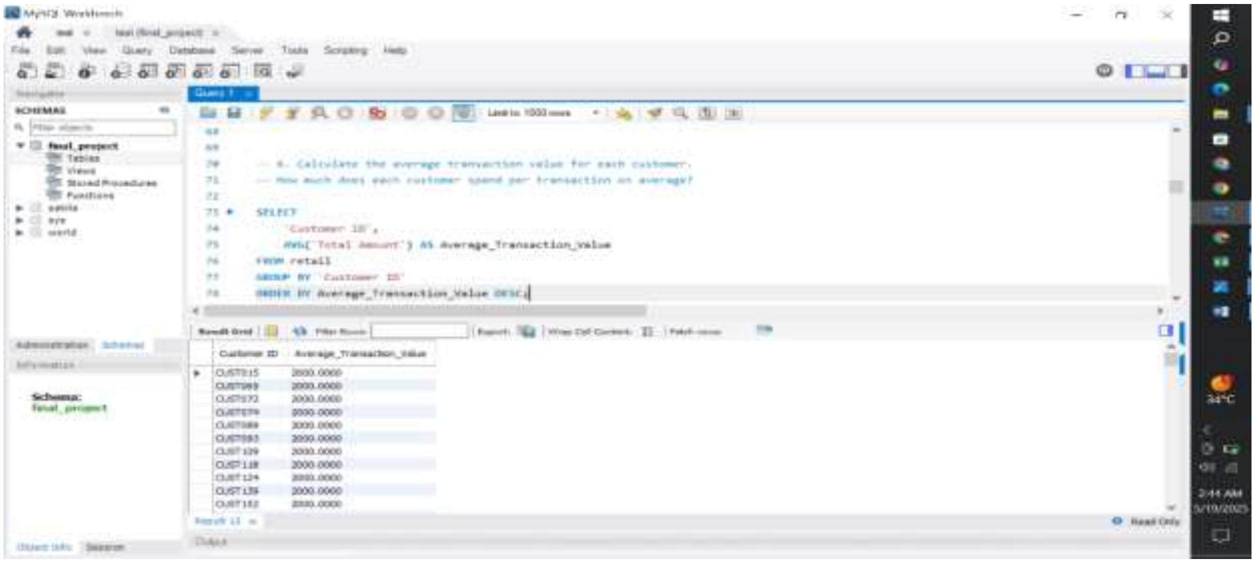
➢ Objective

The goal is to rank the highest spenders to target for loyalty programs or promotions.

6. Calculate the average transaction value for each customer. How much does each customer spend per transaction on average?

```
SELECT
   `Customer ID`,
   AVG(`Total Amount`) AS Average Transaction Value
FROM retail
GROUP BY `Customer ID`
```

ORDER BY Average Transaction Value DESC;



```
-- 6. Calculate the average transaction value for each customer.
-- How much does each customer spend per transaction on average?

SELECT
    `Customer ID`,
    AVG(`Total Amount`) AS Average_Transaction_Value
FROM retail
GROUP BY `Customer ID`
ORDER BY Average_Transaction_Value DESC;
```

```
Customer ID   Average_Transaction_Value
CUST015       2000.0000
CUST069       2000.0000
CUST072       2000.0000
CUST079       2000.0000
CUST088       2000.0000
CUST093       2000.0000
CUST109       2000.0000
CUST118       2000.0000
CUST124       2000.0000
CUST139       2000.0000
CUST182       2000.0000
```

➢ Objective

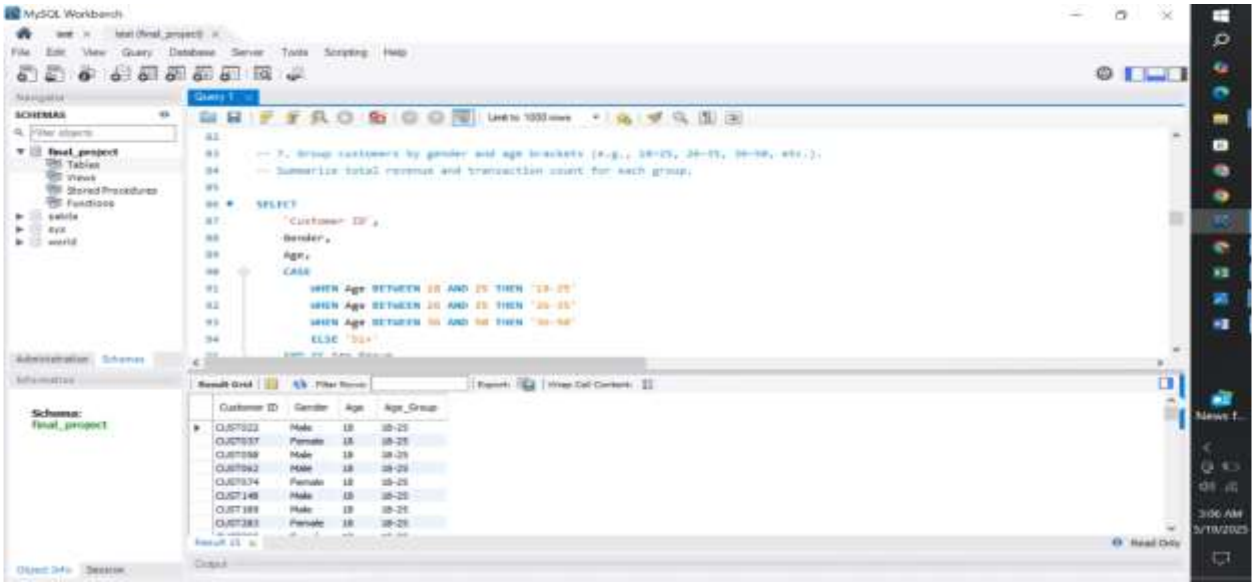
Understand typical spending behavior to guide pricing and bundle strategies.

7. Group customers by gender and age brackets (e.g., 18–25, 26–35, 36–50, etc.). Summarize total revenue and transaction count for each group.

```
SELECT
   `Customer ID`,
   Gender,
   Age,
   CASE
      WHEN Age BETWEEN 18 AND 25 THEN '18-25'
      WHEN Age BETWEEN 26 AND 35 THEN '26-35'
      WHEN Age BETWEEN 36 AND 50 THEN '36-50'
      ELSE '51+'
   END AS Age Group
FROM retail
WHERE Age BETWEEN 18 AND 50
```

GROUP BY `Customer ID`, Gender, Age

ORDER BY Age;



```
-- 7. Group customers by gender and age brackets (e.g., 18-25, 26-35, 36-50, etc.).
-- Summarize total revenue and transaction count for each group.

SELECT
    `Customer ID`,
    Gender,
    Age,
    CASE
        WHEN Age BETWEEN 18 AND 25 THEN '18-25'
        WHEN Age BETWEEN 26 AND 35 THEN '26-35'
        WHEN Age BETWEEN 36 AND 50 THEN '36-50'
        ELSE '51+'
```

Customer ID	Gender	Age	Age_Group
CUST022	Male	18	18-25
CUST037	Female	18	18-25
CUST058	Male	18	18-25
CUST062	Male	18	18-25
CUST074	Female	18	18-25
CUST148	Male	18	18-25
CUST189	Male	18	18-25
CUST283	Female	18	18-25

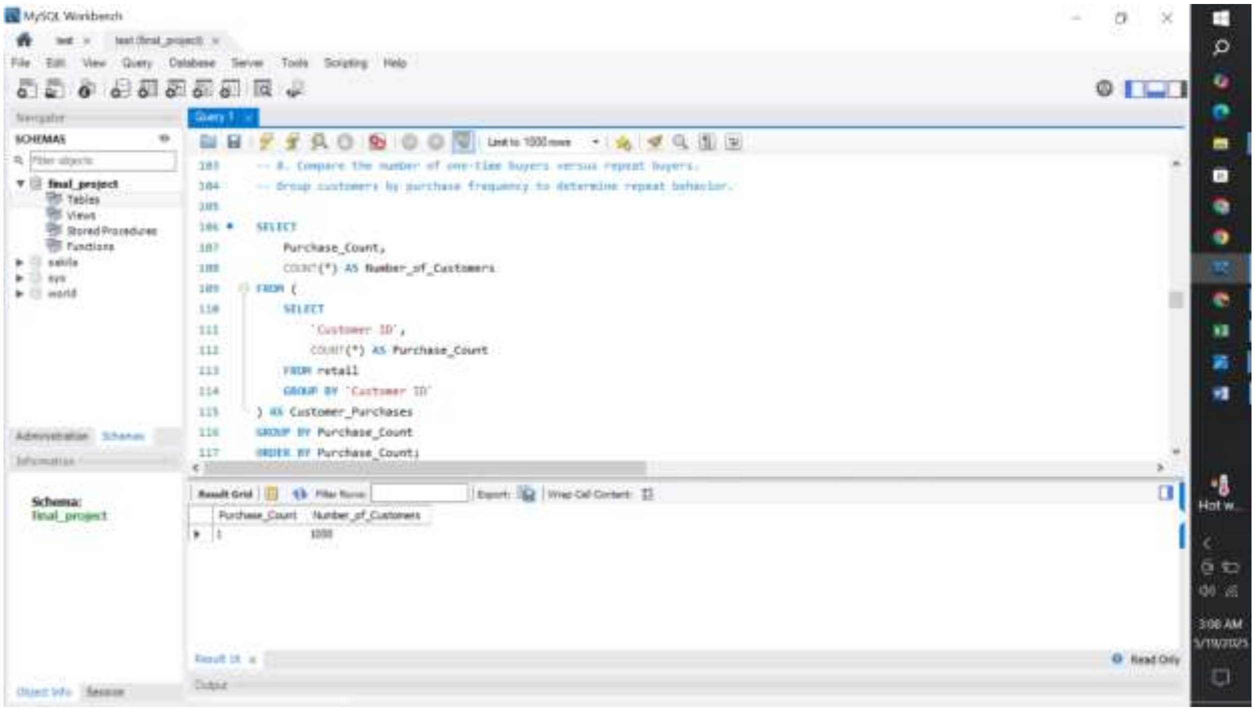
➢ Objective

Segment the customer base for demographic-based marketing and outreach.

8. Compare the number of one-time buyers versus repeat buyers. Group customers by purchase frequency to determine repeat behavior.

```
SELECT
    Purchase Count,
    COUNT(*) AS Number of Customers
FROM (
    SELECT
        `Customer ID`,
        COUNT(*) AS Purchase Count
    FROM retail
    GROUP BY `Customer ID`
) AS Customer Purchases
GROUP BY Purchase Count
```

ORDER BY Purchase Count;



```
-- 8. Compare the number of one-time buyers versus repeat buyers.
-- Group customers by purchase frequency to determine repeat behavior.

SELECT
    Purchase_Count,
    COUNT(*) AS Number_of_Customers
FROM (
    SELECT
        `Customer ID`,
        COUNT(*) AS Purchase_Count
    FROM retail
    GROUP BY `Customer ID`
) AS Customer_Purchases
GROUP BY Purchase_Count
ORDER BY Purchase_Count;
```

```
Purchase_Count  Number_of_Customers
1               1000
```

- **Objective**

Segment the customer base for demographic-based marketing and outreach.

9. Identify inactive customers who have not made a purchase in the last 6 months. Use the most recent date in the dataset as the reference point.

SELECT

`Customer ID`,

MAX(Date) AS Last Purchase Date

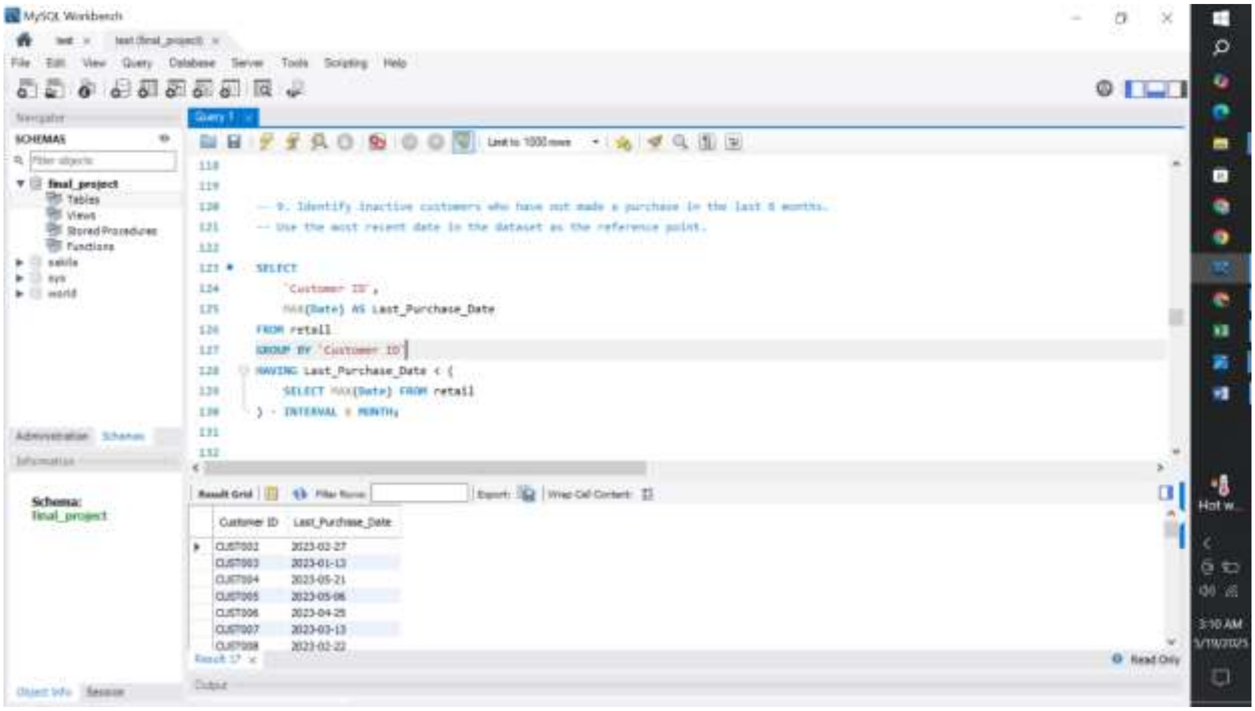
FROM retail

GROUP BY `Customer ID`

HAVING Last Purchase Date < (

SELECT MAX(Date) FROM retail

) - INTERVAL 6 MONTH;



```
-- 9. Identify inactive customers who have not made a purchase in the last 6 months.
-- Use the most recent date in the dataset as the reference point.

SELECT
    `Customer ID`,
    MAX(Date) AS Last_Purchase_Date
FROM retail
GROUP BY `Customer ID`
HAVING Last_Purchase_Date < (
    SELECT MAX(Date) FROM retail
) - INTERVAL 6 MONTH;
```

Customer ID	Last_Purchase_Date
CUST002	2023-02-27
CUST003	2023-01-13
CUST004	2023-05-21
CUST005	2023-05-06
CUST006	2023-04-25
CUST007	2023-03-13
CUST008	2023-02-22

➢ Objective

Spot customers who haven’t purchased in the last 6 months for re-engagement campaigns.

10-- 10. Perform RFM (Recency, Frequency, Monetary) analysis for customer segmentation. Recency: Days since last purchase; Frequency: Number of purchases; Monetary: Total amount spent.

-- Step 1: Calculate RFM metrics for each customer

SELECT

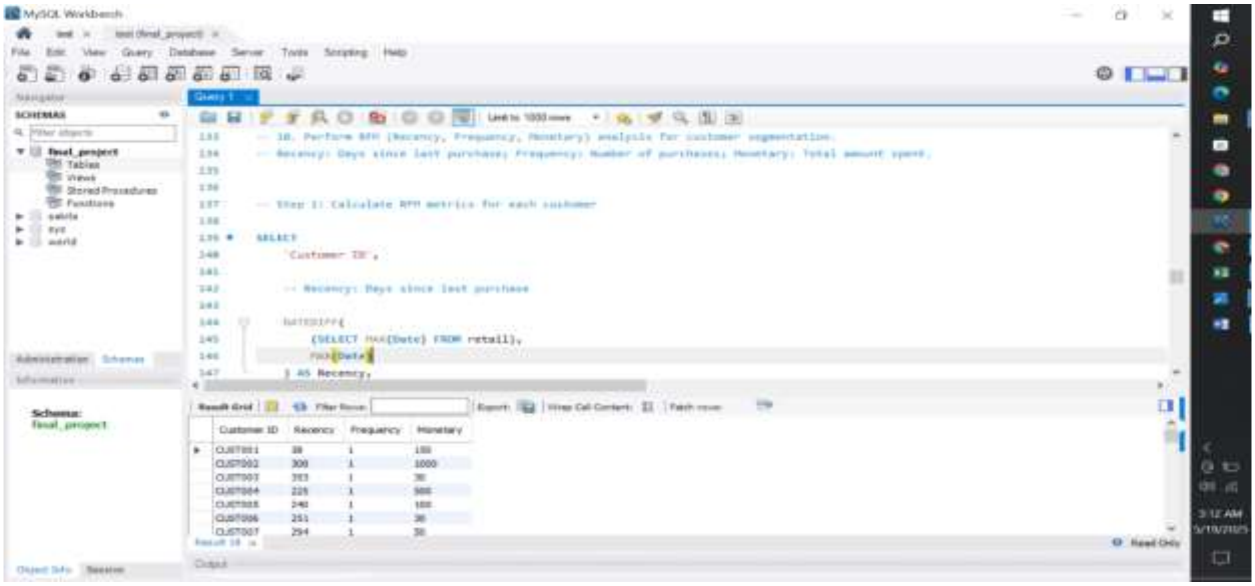
`Customer ID`,

-- Recency: Days since last purchase

DATEDIFF(

(SELECT MAX(Date) FROM retail),

```
        MAX(Date)
    ) AS Recency,

    -- Frequency: Number of purchases

    COUNT(*) AS Frequency,

    -- Monetary: Total amount spent

    SUM(`Total Amount`) AS Monetary
FROM retail
GROUP BY `Customer ID`
ORDER BY `Customer ID`;
```



```
-- 10. Perform RFM (Recency, Frequency, Monetary) analysis for customer segmentation.
-- Recency: Days since last purchase; Frequency: Number of purchases; Monetary: Total amount spent.


-- Step 1: Calculate RFM metrics for each customer

SELECT
    `Customer ID`,

    -- Recency: Days since last purchase

    DATEDIFF(
        (SELECT MAX(Date) FROM retail),
        MAX(Date)
    ) AS Recency,
```

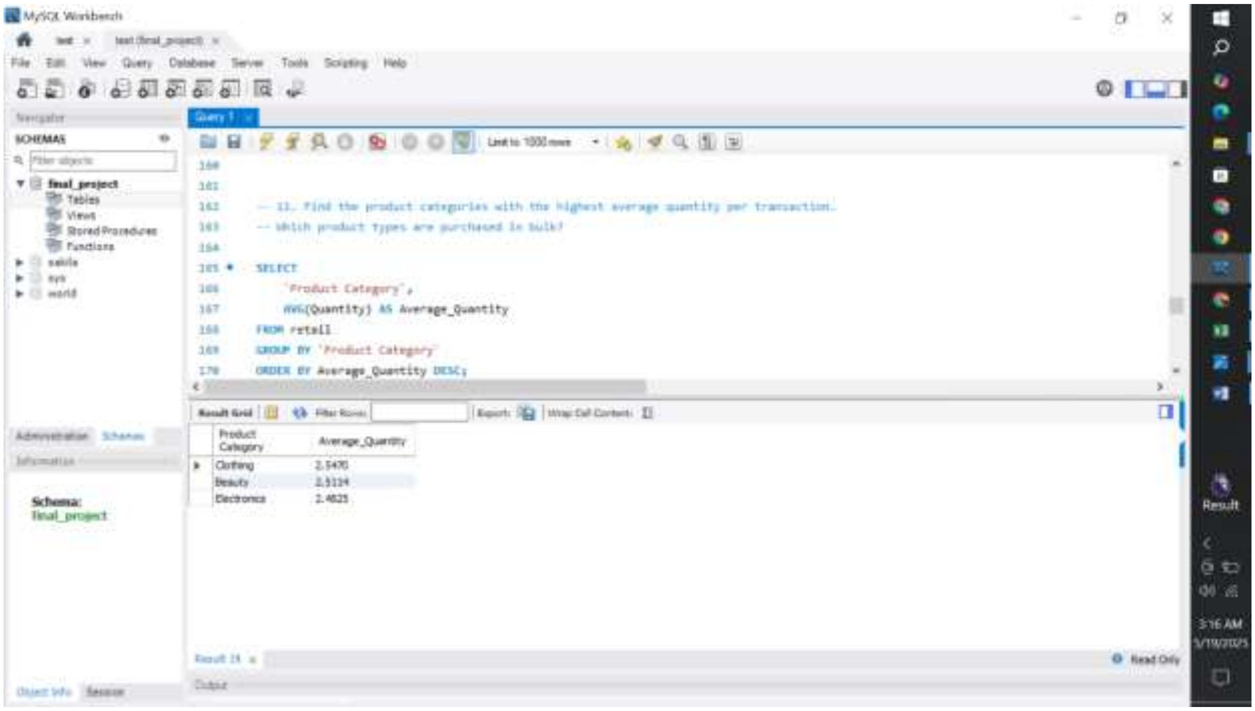
```
   Customer ID   Recency   Frequency   Monetary
▶  CUST001       38        1           150
   CUST002       300       1           1000
   CUST003       353       1           30
   CUST004       225       1           500
   CUST005       240       1           100
   CUST006       251       1           30
   CUST007       294       1           50
```

➢ Objective

Classify customers based on recency, frequency, and monetary value for targeted strategy.

11. Find the product categories with the highest average quantity per transaction. Which product types are purchased in bulk?

```
SELECT
   `Product Category`,
   AVG(Quantity) AS Average Quantity
FROM retail
GROUP BY `Product Category`
ORDER BY Average Quantity DESC;
```



```
-- 11. Find the product categories with the highest average quantity per transaction.
-- Which product types are purchased in bulk?

SELECT
    `Product Category`,
    AVG(Quantity) AS Average_Quantity
FROM retail
GROUP BY `Product Category`
ORDER BY Average_Quantity DESC;
```

Product Category	Average_Quantity
Clothing	2.5470
Beauty	2.5114
Electronics	2.4825

- ## Objective

Identify which categories are typically bought in larger quantities.

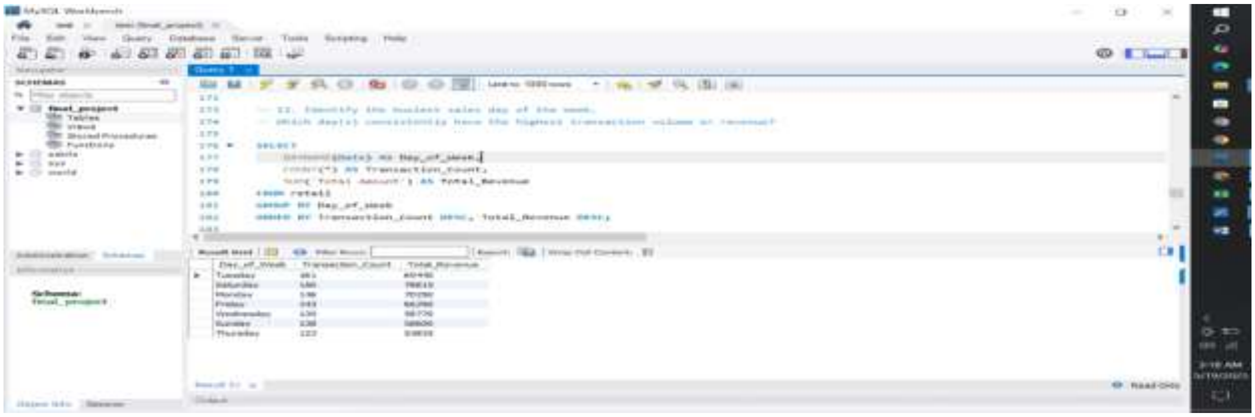
12. Identify the busiest sales day of the week. Which day(s) consistently have the highest transaction volume or revenue?

```
SELECT
   DAYNAME(Date) AS Day of Week,
   COUNT(*) AS Transaction Count,
   SUM(`Total Amount`) AS Total Revenue
```

FROM retail

GROUP BY Day of Week

ORDER BY Transaction Count DESC, Total Revenue DESC;



```
-- 12. Identify the busiest sales day of the week.
-- (Which day(s) consistently have the highest transaction volume or revenue?

SELECT
    DAYNAME(Date) AS Day_of_Week,
    COUNT(*) AS Transaction_Count,
    SUM(`Total Amount`) AS Total_Revenue
FROM retail
GROUP BY Day_of_Week
ORDER BY Transaction_Count DESC, Total_Revenue DESC;
```

Day_of_Week	Transaction_Count	Total_Revenue
Tuesday	[illegible]	[illegible]
Saturday	[illegible]	[illegible]
Monday	[illegible]	[illegible]
Friday	[illegible]	[illegible]
Wednesday	[illegible]	[illegible]
Sunday	[illegible]	[illegible]
Thursday	[illegible]	[illegible]

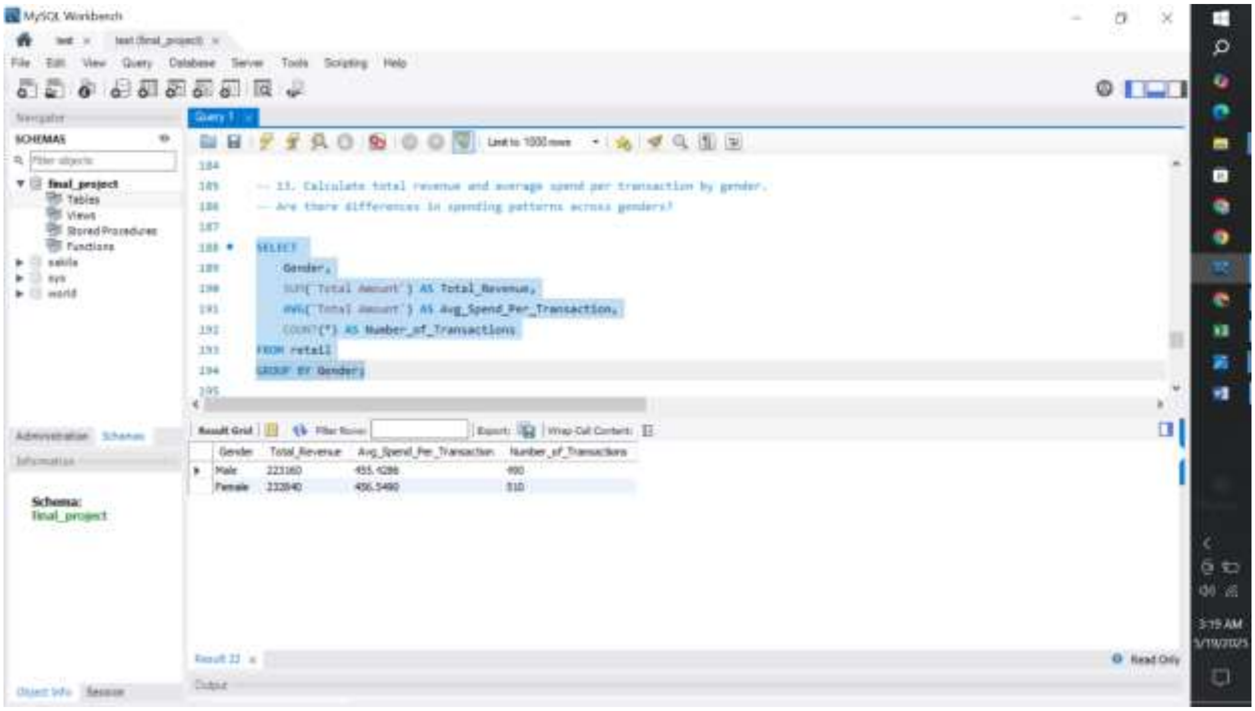
➢ Objective

Discover which days generate the most sales or revenue to optimize staffing and promotions.

13. Calculate total revenue and average spend per transaction by gender. Are there differences in spending patterns across genders?

```
SELECT
   Gender,
   SUM(`Total Amount`) AS Total Revenue,
   AVG(`Total Amount`) AS Avg Spend Per Transaction,
   COUNT(*) AS Number of Transactions
FROM retail
```

GROUP BY Gender;



```
-- 13. Calculate total revenue and average spend per transaction by gender.
-- Are there differences in spending patterns across genders?

SELECT
    Gender,
    SUM(`Total Amount`) AS Total_Revenue,
    AVG(`Total Amount`) AS Avg_Spend_Per_Transaction,
    COUNT(*) AS Number_of_Transactions
FROM retail
GROUP BY Gender;
```

Gender	Total_Revenue	Avg_Spend_Per_Transaction	Number_of_Transactions
Male	223160	455.4286	490
Female	232840	456.5490	510

➢ Objective

Explore how spending differs across male and female customers.

14. Find the top 5 most frequently purchased product categories. Based on number of transactions involving each category.

SELECT

`Product Category`,

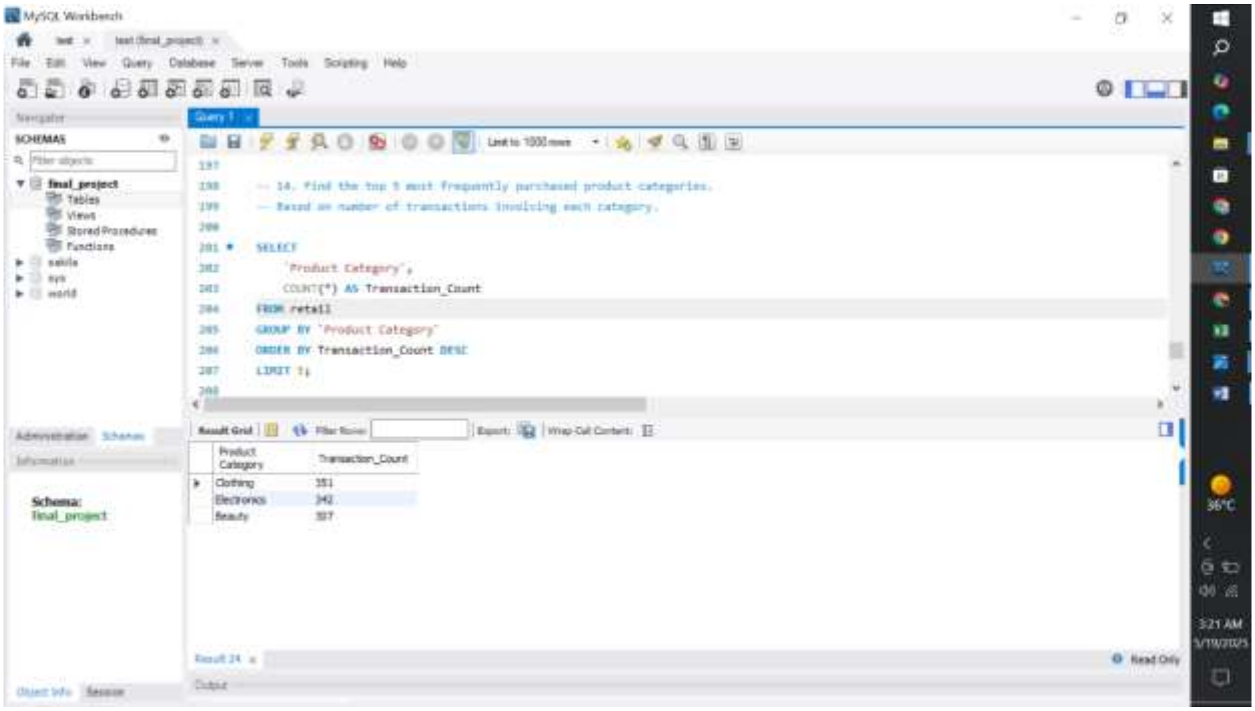
COUNT(*) AS Transaction Count

FROM retail

GROUP BY `Product Category`

ORDER BY Transaction Count DESC

LIMIT 5;



```sql
-- 14. Find the top 5 most frequently purchased product categories.
-- Based on number of transactions involving each category.

SELECT
    `Product Category`,
    COUNT(*) AS Transaction_Count
FROM retail
GROUP BY `Product Category`
ORDER BY Transaction_Count DESC
LIMIT 5;
```

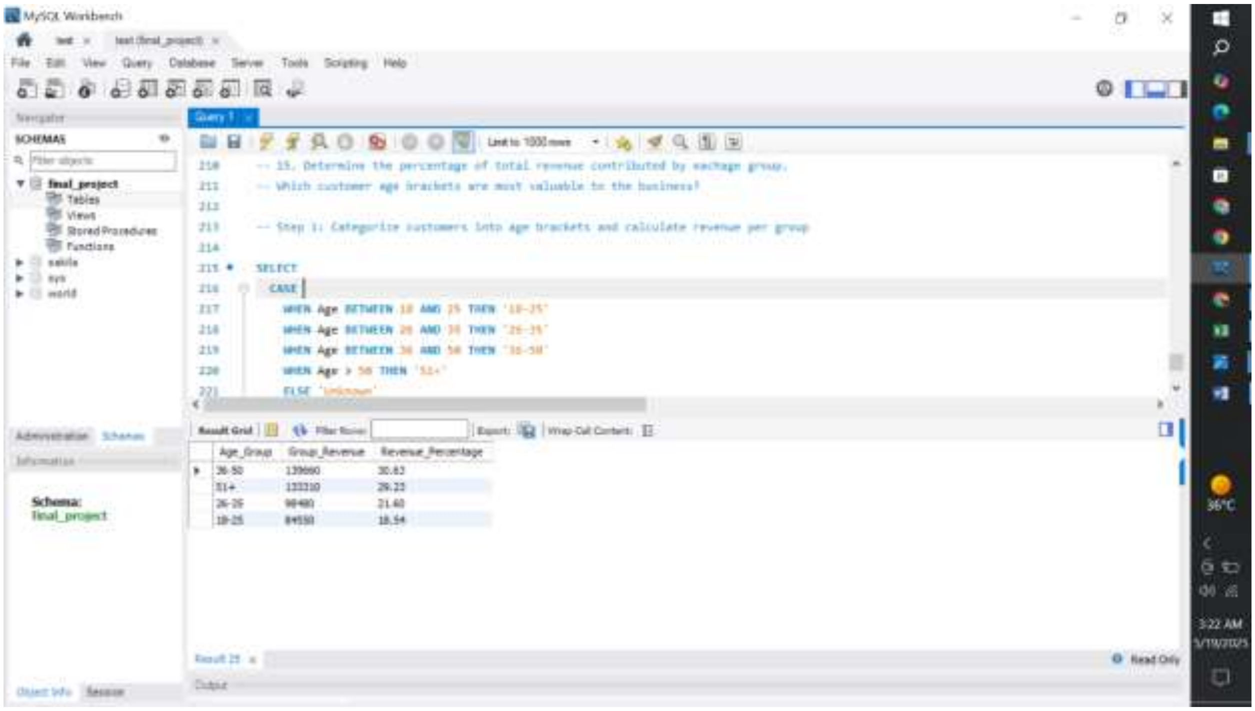
Product Category	Transaction_Count
Clothing	351
Electronics	342
Beauty	307

➢ Objective

Pinpoint popular product categories based on transaction count.

15. Determine the percentage of total revenue contributed by each age group. Which customer age brackets are most valuable to the business?

```
-- Step 1: Categorize customers into age brackets and calculate revenue per group

SELECT
 CASE
  WHEN Age BETWEEN 18 AND 25 THEN '18-25'
  WHEN Age BETWEEN 26 AND 35 THEN '26-35'
  WHEN Age BETWEEN 36 AND 50 THEN '36-50'
  WHEN Age > 50 THEN '51+'
  ELSE 'Unknown'
 END AS Age Group,
  SUM(`Total Amount`) AS Group Revenue,


  ROUND(SUM(`Total Amount`) * 100.0 /
        (SELECT SUM(`Total Amount`) FROM retail), 2) AS Revenue Percentage
FROM retail
GROUP BY Age Group
ORDER BY Revenue Percentage DESC;
```



```
-- 15. Determine the percentage of total revenue contributed by eachage group.
-- Which customer age brackets are most valuable to the business?

-- Step 1: Categorize customers into age brackets and calculate revenue per group

SELECT
  CASE
    WHEN Age BETWEEN 18 AND 25 THEN '18-25'
    WHEN Age BETWEEN 26 AND 35 THEN '26-35'
    WHEN Age BETWEEN 36 AND 50 THEN '36-50'
    WHEN Age > 50 THEN '51+'
    ELSE 'Unknown'
```

Age_Group	Group_Revenue	Revenue_Percentage
36-50	139660	30.63
51+	133310	29.23
26-35	98480	21.60
18-25	84550	18.54

➢ Objective

Highlight which age brackets are most valuable to the business in terms of total spending.