**Student Name:** Ba Huy Hoang Le

**Student ID:** 224309594

**Query Appendix**

# Question 1

* Query 1a:

Finding which products are the most popular by volume (based on Quantity)

SELECT Product\_SKU, SUM(Quantity) AS Total\_Product\_Quantity

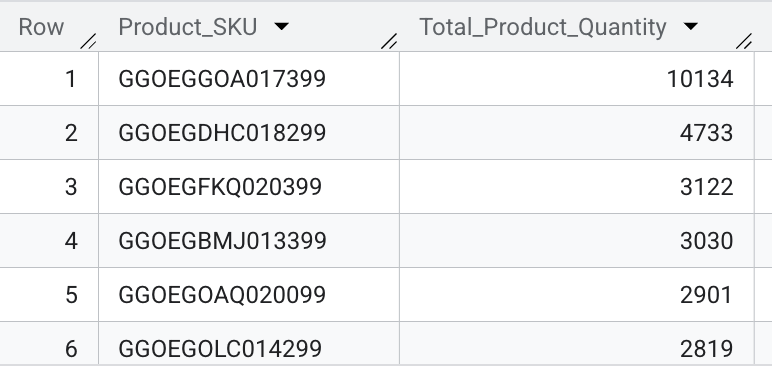
FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

WHERE EXTRACT(YEAR FROM Transaction\_Date) = 2024

GROUP BY Product\_SKU

ORDER BY SUM (Quantity) DESC;

* Result Screenshot



* Finding the number of unique customers who purchased product GGOEGGOA017399, the total number of transactions, and average rating

SELECT

COUNT(DISTINCT Customer\_ID) AS Total\_Customer,

COUNT(Transaction\_ID) AS Total\_Transaction,

ROUND(AVG(Transaction\_Rating),2) AS Average\_Trans\_Rating

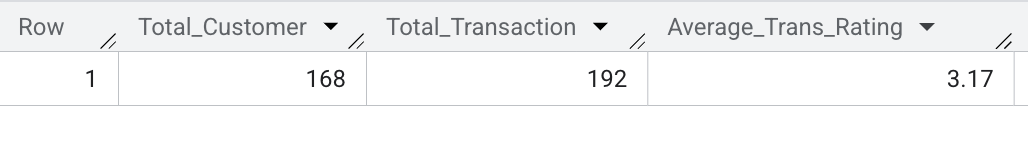
FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

WHERE

Product\_SKU = 'GGOEGGOA017399' AND

EXTRACT(YEAR FROM Transaction\_Date) = 2024;

* Result Screenshot



* Query 1b:

Finding which products are the most popular by reach (based on number of unique customers)

SELECT Product\_SKU, COUNT(DISTINCT Customer\_ID) AS Total\_Unique\_Customer

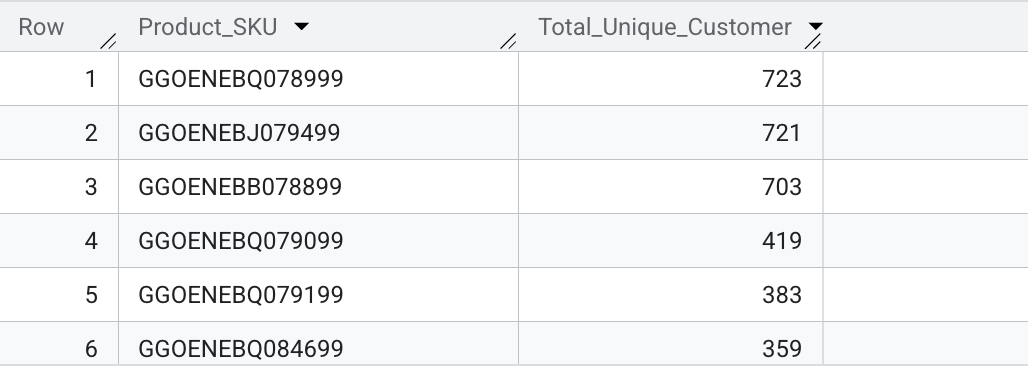
FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

WHERE EXTRACT(YEAR FROM Transaction\_Date) = 2024

GROUP BY Product\_SKU

ORDER BY Total\_Unique\_Customer DESC;

* Result Screenshot



Finding the number of unique customers who purchased product GGOENEBQ078999, the total number of transactions, and average rating

SELECT

COUNT(DISTINCT Customer\_ID) AS Total\_Customer,

COUNT(Transaction\_ID) AS Total\_Transaction,

ROUND(AVG(Transaction\_Rating),2) AS Average\_Trans\_Rating

FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

WHERE

Product\_SKU = 'GGOENEBQ078999' AND

EXTRACT(YEAR FROM Transaction\_Date) = 2024;

* Result Screenshot

A screenshot of a phone

AI-generated content may be incorrect.

* Query 1c:

Finding which products are the most popular number of transactions (Transaction\_ID)

SELECT Product\_SKU, COUNT(Transaction\_ID) AS Total\_Transaction

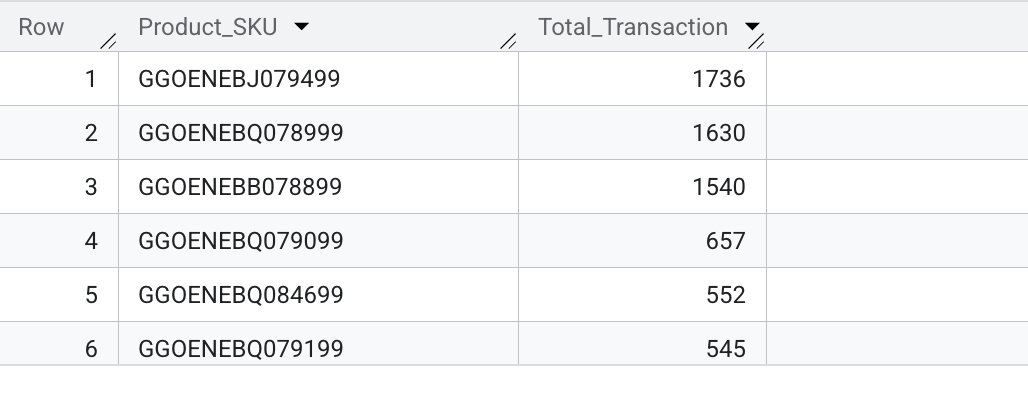
FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

WHERE EXTRACT(YEAR FROM Transaction\_Date) = 2024

GROUP BY Product\_SKU

ORDER BY Total\_Transaction DESC;

* Result Screenshot



Finding the number of unique customers who purchased product GGOENEBJ079499, the total number of transactions, and average rating

SELECT

COUNT(DISTINCT Customer\_ID) AS Total\_Customer,

COUNT(Transaction\_ID) AS Total\_Transaction,

ROUND(AVG(Transaction\_Rating),2) AS Average\_Trans\_Rating

FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

WHERE

Product\_SKU = 'GGOENEBJ079499' AND

EXTRACT(YEAR FROM Transaction\_Date) = 2024;

* Result Screenshot

A screenshot of a phone

AI-generated content may be incorrect.

* Query 1d:

Finding which products are most favored

SELECT Product\_SKU, ROUND(AVG(Transaction\_Rating),2) AS Average\_Trans\_Rating

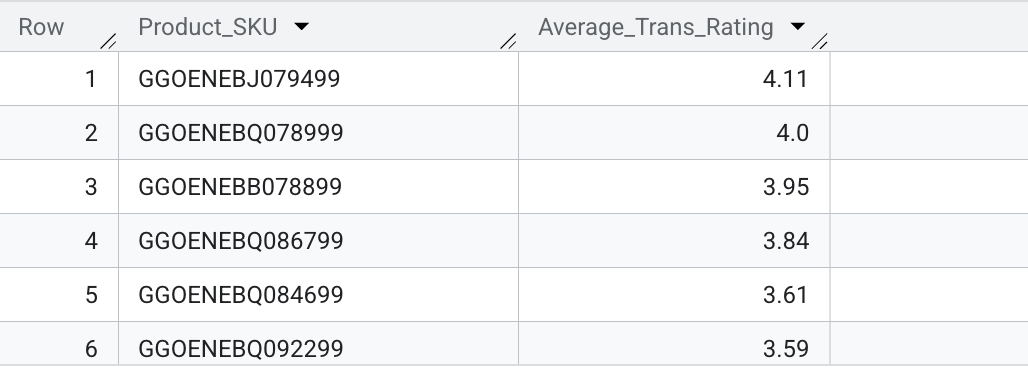
FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

WHERE EXTRACT(YEAR FROM Transaction\_Date) = 2024

GROUP BY Product\_SKU

ORDER BY Average\_Trans\_Rating DESC;

* Result Screenshot



Finding the number of unique customers who purchased product GGOENEBJ079499, the total number of transactions, and average rating

SELECT

COUNT(DISTINCT Customer\_ID) AS Total\_Customer,

COUNT(Transaction\_ID) AS Total\_Transaction,

ROUND(AVG(Transaction\_Rating),2) AS Average\_Trans\_Rating

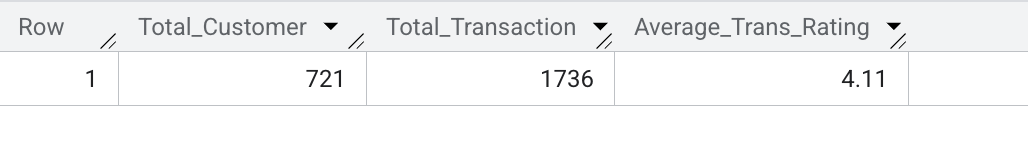
FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

WHERE

Product\_SKU = 'GGOENEBJ079499' AND

EXTRACT(YEAR FROM Transaction\_Date) = 2024;

* Result Screenshot



# Question 2

* Query 2a:

Calculating Recency, Frequency, and Monetary Score

SELECT

Customer\_ID,

DATE\_DIFF(CURRENT\_DATE(), MAX(Transaction\_Date), DAY) AS Recency\_Score,

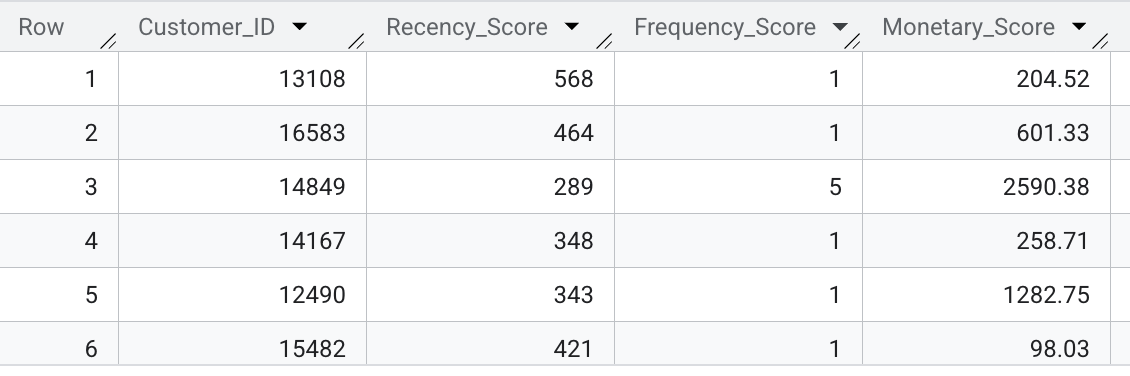
COUNT(DISTINCT Transaction\_Date) AS Frequency\_Score,

ROUND(SUM(Avg\_Price \* Quantity \* (1 - IFNULL(Discount\_pct, 0)/100)), 2) AS Monetary\_Score

FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

GROUP BY Customer\_ID;

* Result Screenshot



* Query 2b:

Finding the most valuable customers based on Recency\_Score

SELECT

Customer\_ID,

DATE\_DIFF(CURRENT\_DATE(), MAX(Transaction\_Date), DAY) AS Recency\_Score

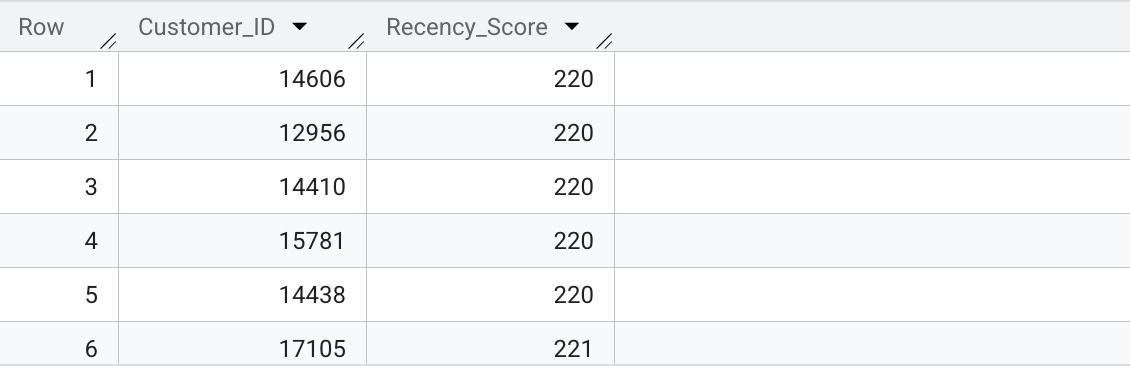
FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

GROUP BY Customer\_ID

ORDER BY Recency\_Score ASC

LIMIT 100;

* Result Screenshot



* Query 2c:

Finding the most valuable customers based on Frequency\_Score

SELECT

Customer\_ID,

DATE\_DIFF(CURRENT\_DATE(), MAX(Transaction\_Date), DAY) AS Recency\_Score

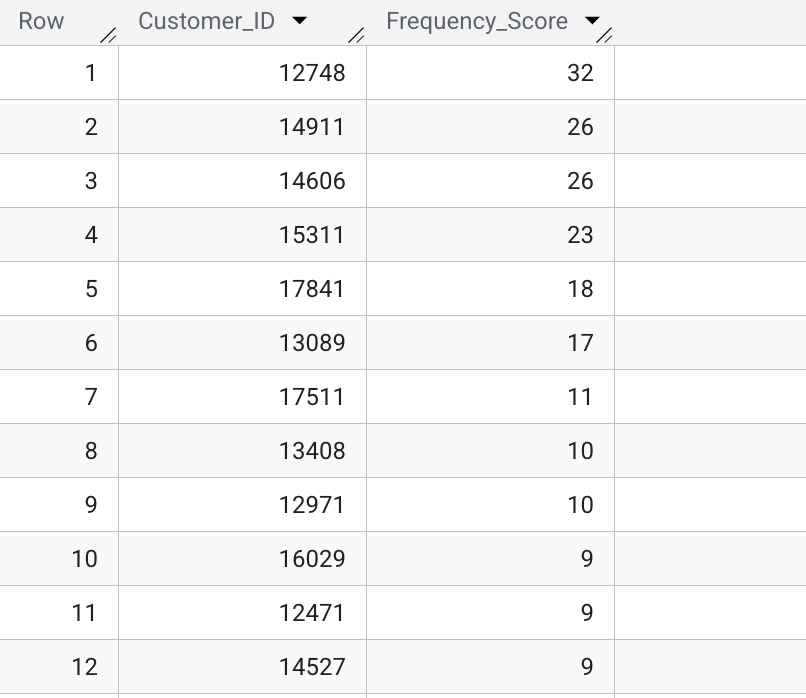
FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

GROUP BY Customer\_ID

ORDER BY Recency\_Score ASC

LIMIT 100;

* Result Screenshot



* Query 2d:

Finding most valuable customers based on Monetary\_Score

SELECT

Customer\_ID,

ROUND(SUM(Avg\_Price \* Quantity \* (1 - IFNULL(Discount\_pct, 0)/100)), 2) AS Monetary\_Score

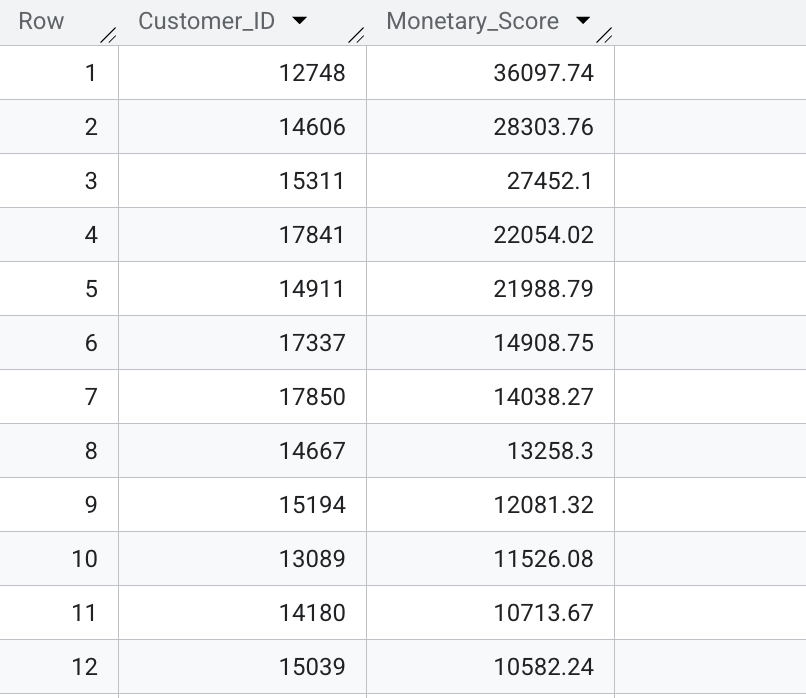
FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

GROUP BY Customer\_ID

ORDER BY Monetary\_Score DESC

LIMIT 100;

* Result Screenshot



# Question 3

* Query 3a:

Segment customers into five groups for each of the three key metrics: Recency, Frequency, and Monetary value based on their transaction history.

WITH RFM\_Scores AS (

SELECT

Customer\_ID,

DATE\_DIFF(CURRENT\_DATE(), MAX(Transaction\_Date), DAY) AS Recency\_Score,

COUNT(DISTINCT Transaction\_Date) AS Frequency\_Score,

ROUND(SUM(Avg\_Price \* Quantity \* (1 - IFNULL(Discount\_pct, 0)/100)), 2) AS Monetary\_Score

FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

GROUP BY Customer\_ID

),

RFM\_Quintiles AS (

SELECT

Customer\_ID,

NTILE(5) OVER (ORDER BY Recency\_Score DESC, Customer\_ID) AS Recency\_Quintile,

NTILE(5) OVER (ORDER BY Frequency\_Score ASC, Customer\_ID) AS Frequency\_Quintile,

NTILE(5) OVER (ORDER BY Monetary\_Score ASC, Customer\_ID) AS Monetary\_Quintile

FROM RFM\_Scores

)

SELECT

Customer\_ID,

Recency\_Quintile,

Frequency\_Quintile,

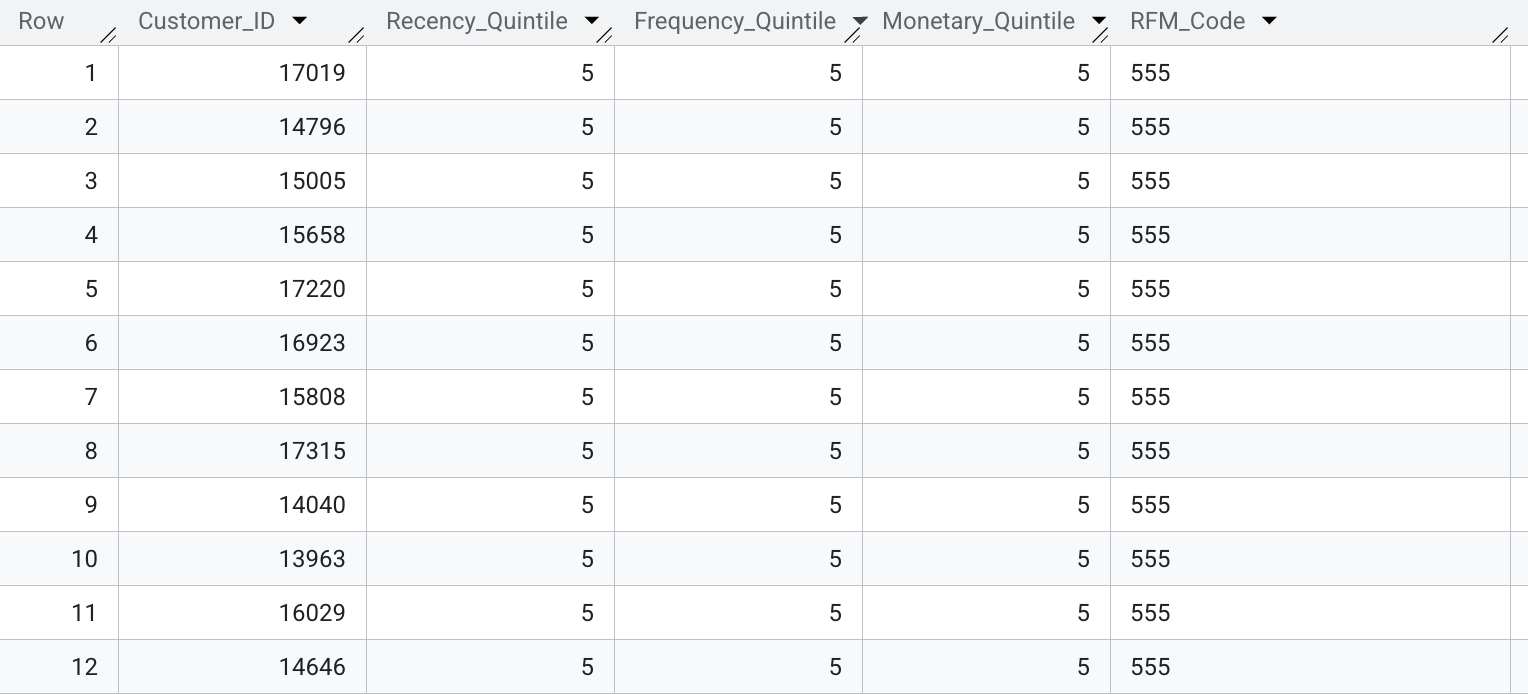
Monetary\_Quintile,

CONCAT(Recency\_Quintile, Frequency\_Quintile, Monetary\_Quintile) AS RFM\_Code

FROM RFM\_Quintiles

ORDER BY RFM\_Code DESC;

* Result Screenshot



* Query 3b:

Extra step to name customer segments

WITH RFM\_Scores AS (

SELECT

Customer\_ID,

DATE\_DIFF(CURRENT\_DATE(), MAX(Transaction\_Date), DAY) AS Recency\_Score,

COUNT(DISTINCT Transaction\_Date) AS Frequency\_Score,

ROUND(SUM(Avg\_Price \* Quantity \* (1 - IFNULL(Discount\_pct, 0)/100)), 2) AS Monetary\_Score

FROM `mis784t22025-466123.224309594\_Assign1.Transaction\_Table`

GROUP BY Customer\_ID

),

RFM\_Quintiles AS (

SELECT

Customer\_ID,

NTILE(5) OVER (ORDER BY Recency\_Score DESC, Customer\_ID) AS Recency\_Quintile,

NTILE(5) OVER (ORDER BY Frequency\_Score ASC, Customer\_ID) AS Frequency\_Quintile,

NTILE(5) OVER (ORDER BY Monetary\_Score ASC, Customer\_ID) AS Monetary\_Quintile

FROM RFM\_Scores

),

RFM\_Labeled AS (

SELECT

Customer\_ID,

Recency\_Quintile,

Frequency\_Quintile,

Monetary\_Quintile,

Monetary\_Score,

CASE

WHEN Recency\_Quintile = 5 AND Frequency\_Quintile = 5 AND Monetary\_Quintile = 5 THEN 'Power Users'

WHEN Recency\_Quintile = 5 THEN 'Newly Engaged Buyers'

WHEN Frequency\_Quintile = 5 THEN 'Frequent Shoppers'

WHEN Monetary\_Quintile = 5 THEN 'High-Value Purchasers'

WHEN Recency\_Quintile = 1 THEN 'Lapsed Buyers'

WHEN Frequency\_Quintile = 1 THEN 'Infrequent Buyers'

WHEN Monetary\_Quintile = 1 THEN 'Budget Buyers'

ELSE 'General Segment'

END AS Segment

FROM RFM\_Quintiles

JOIN RFM\_Scores USING (Customer\_ID)

)

SELECT

Segment,

COUNT(\*) AS Num\_Customers,

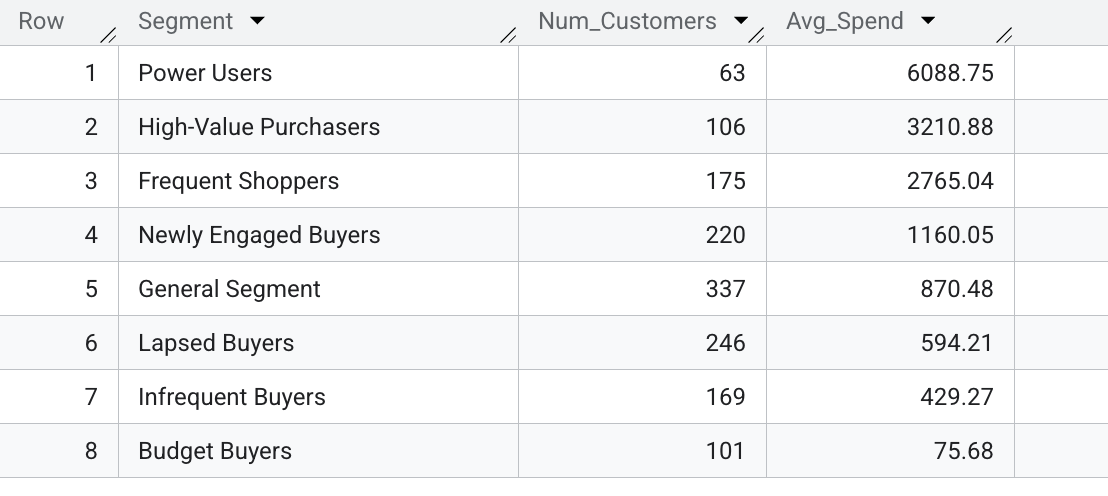
ROUND(AVG(Monetary\_Score), 2) AS Avg\_Spend

FROM RFM\_Labeled

GROUP BY Segment

ORDER BY Avg\_Spend DESC;

* Result Screenshot



# Question 4 - Which types of customer engagement are most strongly associated with higher spending? How do combinations of these behaviors identify SmartStream’s most valuable customers?"

* Query 4a:

CREATE OR REPLACE VIEW `mis784t22025-466123.224309594\_Assign1.Customer\_1\_Engagement\_Monetary` AS

SELECT

c.Customer\_ID,

ROUND(SUM(t.Avg\_Price \* t.Quantity \* (1 - IFNULL(t.Discount\_pct, 0)/100)), 2) AS Monetary\_Score,

c.Chatbot\_Usage\_Count,

c.Email\_Opened\_Count,

c.Clicked\_Ad\_Campaigns,

c.Preferred\_Channel

FROM

`mis784t22025-466123.224309594\_Assign1.Customer\_1` c

INNER JOIN

`mis784t22025-466123.224309594\_Assign1.Transaction\_Table` t

ON c.Customer\_ID = t.Customer\_ID

GROUP BY

c.Customer\_ID,

c.Chatbot\_Usage\_Count,

c.Email\_Opened\_Count,

c.Clicked\_Ad\_Campaigns,

c.Preferred\_Channel;

* Result Screenshot

# 

* Query 4b:

To understand the distribution of email opens, chatbot usage, and ad clicks and justify "High", "Medium", and "Low" engagement levels

SELECT

MIN(Chatbot\_Usage\_Count) AS Min\_Chatbot,

MAX(Chatbot\_Usage\_Count) AS Max\_Chatbot,

AVG(Chatbot\_Usage\_Count) AS Avg\_Chatbot,

MIN(Email\_Opened\_Count) AS Min\_Email,

MAX(Email\_Opened\_Count) AS Max\_Email,

AVG(Email\_Opened\_Count) AS Avg\_Email,

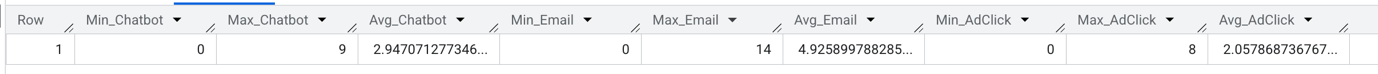
MIN(Clicked\_Ad\_Campaigns) AS Min\_AdClick,

MAX(Clicked\_Ad\_Campaigns) AS Max\_AdClick,

AVG(Clicked\_Ad\_Campaigns) AS Avg\_AdClick

FROM `mis784t22025-466123.224309594\_Assign1.Customer\_1`;

* Result Screenshot



* Query 4c:

To find out which communication channel preference is associated with higher spending

SELECT

Preferred\_Channel,

COUNT(\*) AS Num\_Customers,

ROUND(AVG(Monetary\_Score),2) AS Avg\_Spend

FROM

`mis784t22025-466123.224309594\_Assign1.Customer\_1\_Engagement\_Monetary`

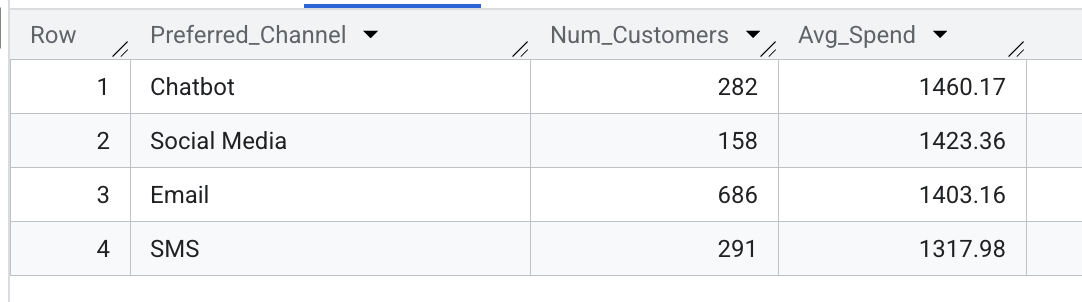
GROUP BY

Preferred\_Channel

ORDER BY

Avg\_Spend DESC;

* Result Screenshot



* Query 4d:

To segments customers by how engaged they are and determine if more engagement means more spending, and which channel is the strongest

SELECT

CASE

WHEN Chatbot\_Usage\_Count = 0 THEN 'None'

WHEN Chatbot\_Usage\_Count <= 1 THEN 'Low'

WHEN Chatbot\_Usage\_Count <= 3 THEN 'Medium'

ELSE 'High'

END AS Chatbot\_Level,

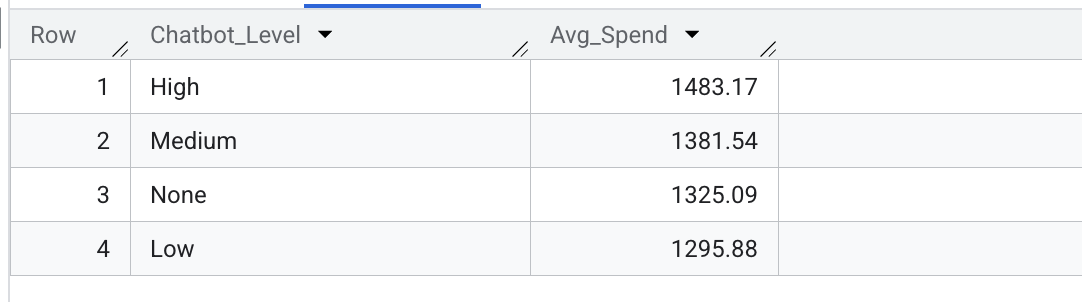
ROUND(AVG(Monetary\_Score), 2) AS Avg\_Spend

FROM `mis784t22025-466123.224309594\_Assign1.Customer\_1\_Engagement\_Monetary`

GROUP BY Chatbot\_Level

ORDER BY Avg\_Spend DESC;

* Result Screenshot



SELECT

CASE

WHEN Email\_Opened\_Count = 0 THEN 'None'

WHEN Email\_Opened\_Count <= 2 THEN 'Low'

WHEN Email\_Opened\_Count <= 5 THEN 'Medium'

ELSE 'High'

END AS Email\_Level,

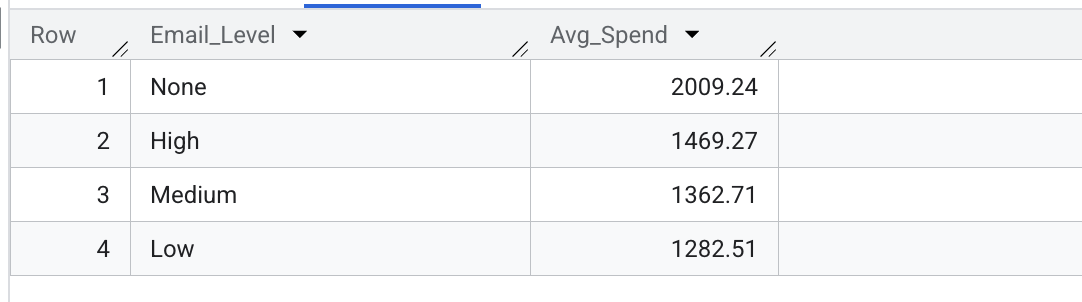
ROUND(AVG(Monetary\_Score), 2) AS Avg\_Spend

FROM `mis784t22025-466123.224309594\_Assign1.Customer\_1\_Engagement\_Monetary`

GROUP BY Email\_Level

ORDER BY Avg\_Spend DESC;

* Result Screenshot



SELECT

CASE

WHEN Clicked\_Ad\_Campaigns = 0 THEN 'None'

WHEN Clicked\_Ad\_Campaigns <= 1 THEN 'Low'

WHEN Clicked\_Ad\_Campaigns <= 2 THEN 'Medium'

ELSE 'High'

END AS Ad\_Level,

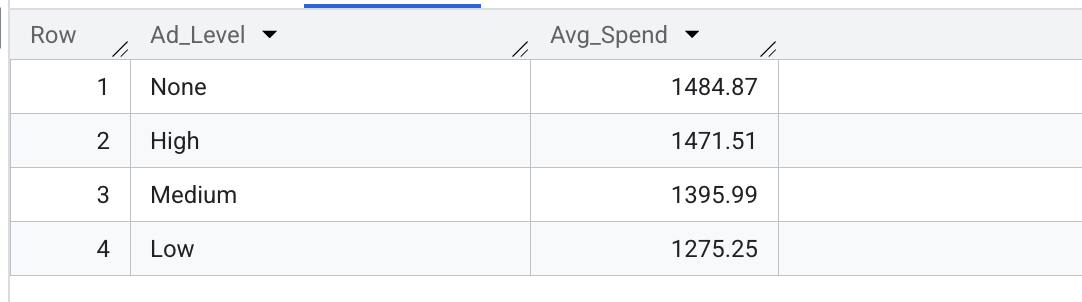
ROUND(AVG(Monetary\_Score), 2) AS Avg\_Spend

FROM `mis784t22025-466123.224309594\_Assign1.Customer\_1\_Engagement\_Monetary`

GROUP BY Ad\_Level

ORDER BY Avg\_Spend DESC;

* Result Screenshot



* Query 4e:

To define targetable customer profiles

WITH Engagement\_Monetary AS (

SELECT

c.Customer\_ID,

ROUND(SUM(t.Avg\_Price \* t.Quantity \* (1 - IFNULL(t.Discount\_pct, 0)/100)), 2) AS Monetary\_Score,

c.Chatbot\_Usage\_Count,

c.Email\_Opened\_Count,

c.Clicked\_Ad\_Campaigns,

c.Preferred\_Channel

FROM

`mis784t22025-466123.224309594\_Assign1.Customer\_1` c

INNER JOIN

`mis784t22025-466123.224309594\_Assign1.Transaction\_Table` t

ON c.Customer\_ID = t.Customer\_ID

GROUP BY

c.Customer\_ID,

c.Chatbot\_Usage\_Count,

c.Email\_Opened\_Count,

c.Clicked\_Ad\_Campaigns,

c.Preferred\_Channel

),

Percentile\_Threshold AS (

SELECT

APPROX\_QUANTILES(Monetary\_Score, 4)[OFFSET(3)] AS Top25\_Threshold

FROM Engagement\_Monetary

),

Flagged AS (

SELECT

e.\*,

CASE WHEN e.Monetary\_Score >= p.Top25\_Threshold THEN 1 ELSE 0 END AS Is\_High\_Spender,

CASE WHEN e.Chatbot\_Usage\_Count > 3 THEN 1 ELSE 0 END AS High\_Chatbot,

CASE WHEN e.Email\_Opened\_Count > 5 THEN 1 ELSE 0 END AS High\_Email,

CASE WHEN e.Clicked\_Ad\_Campaigns > 2 THEN 1 ELSE 0 END AS High\_AdClick

FROM Engagement\_Monetary e

CROSS JOIN Percentile\_Threshold p

)

SELECT

Is\_High\_Spender,

High\_Chatbot,

High\_Email,

High\_AdClick,

COUNT(\*) AS Num\_Customers

FROM Flagged

GROUP BY

Is\_High\_Spender,

High\_Chatbot,

High\_Email,

High\_AdClick

ORDER BY

Num\_Customers DESC;

* Result Screenshot

