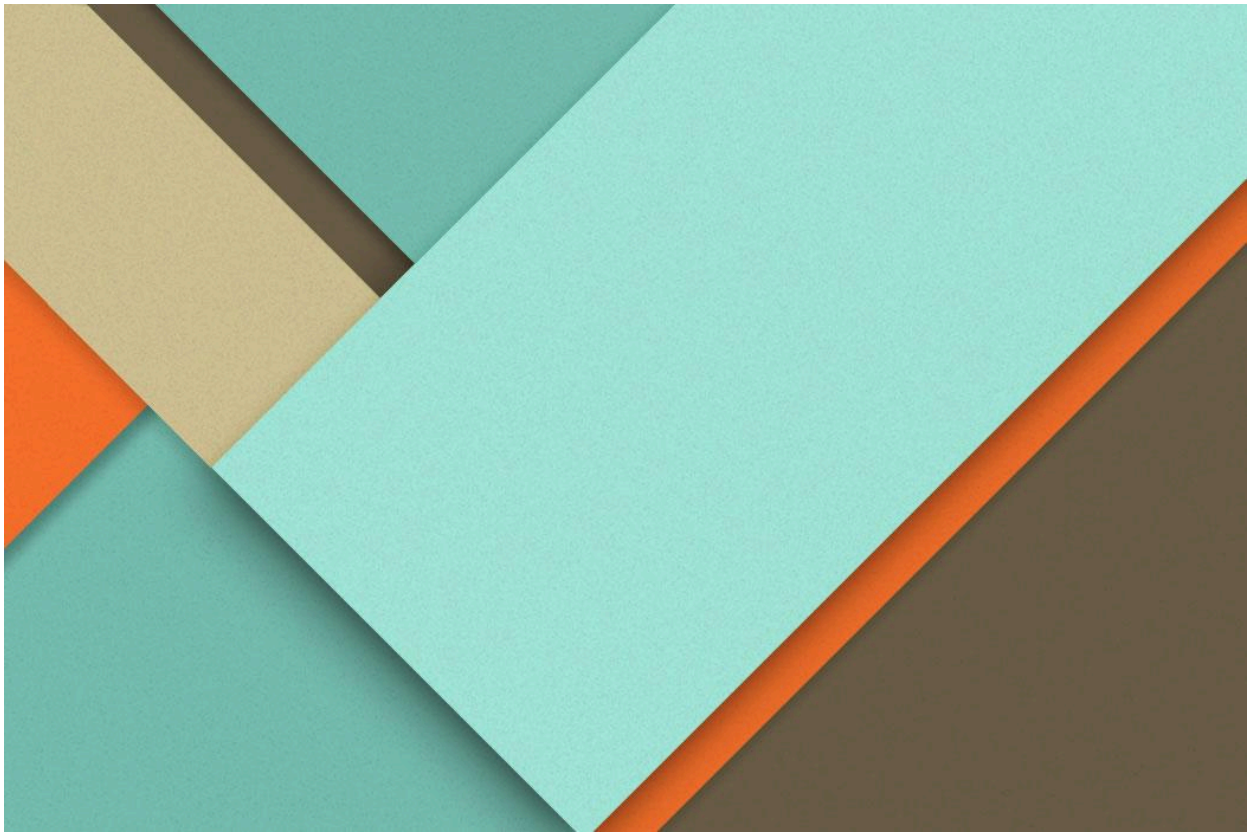


TARGET (SQL)

Introduction

● Target is a globally renowned brand and a prominent retailer in the United States. Target makes itself a preferred shopping destination by offering outstanding value, inspiration, innovation and an exceptional guest experience that no other retailer can deliver.

● This particular business case focuses on the operations of Target in Brazil and provides insightful information about 100,000 orders placed between 2016 and 2018. The dataset offers a comprehensive view of various dimensions including the order status, price, payment and freight performance, customer location, product attributes, and customer reviews.

● By analyzing this extensive dataset, it becomes possible to gain valuable insights into Target's operations in Brazil. The information can shed light on various aspects of the business, such as order processing, pricing strategies, payment and shipping efficiency.

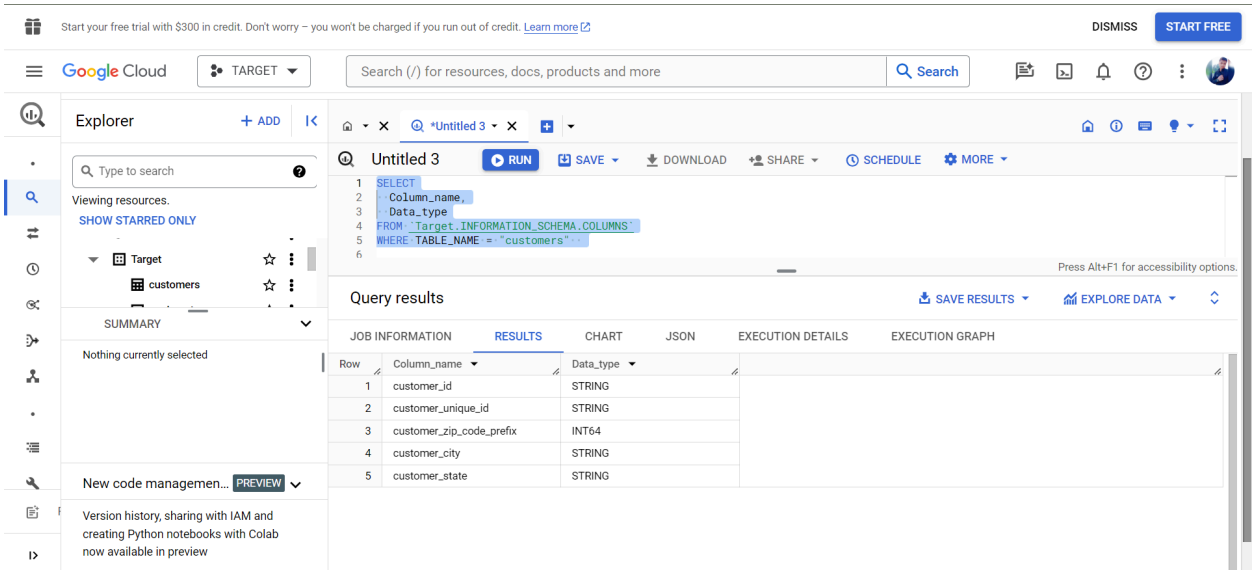
I. Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset.

A. Data type of all columns in the “customers” table.

Query:

```
SELECT
  Column_name,
  Data_type
FROM `Target.INFORMATION_SCHEMA.COLUMNS`
WHERE TABLE_NAME = "customers"
```

Output screenshot:



Insights:

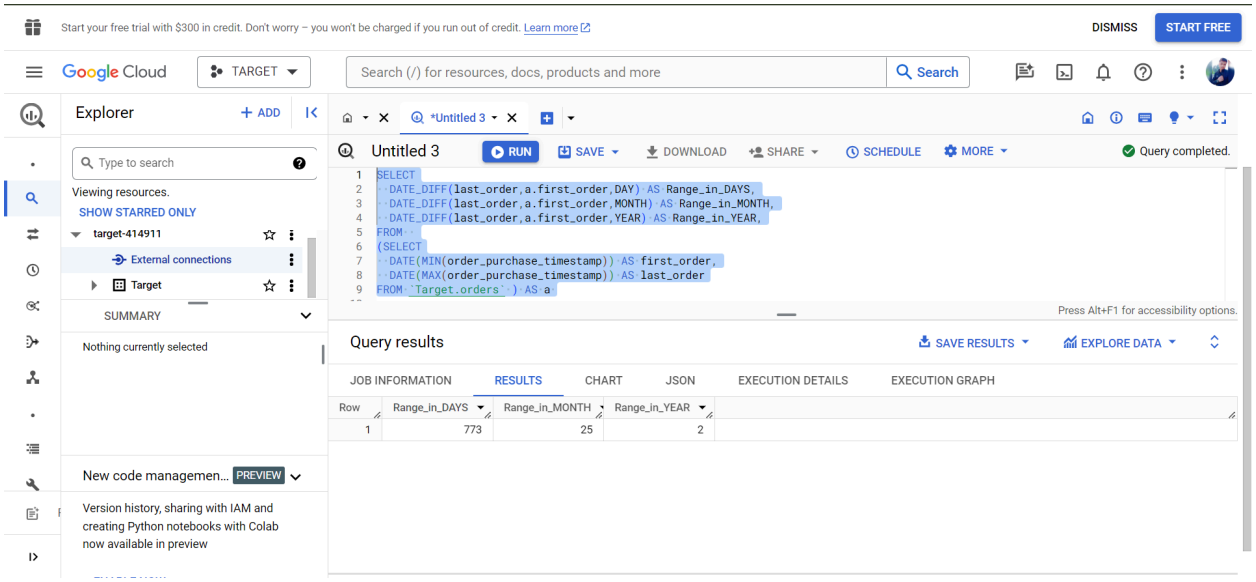
- All the data type is “STRING” except customer_zip_code_prefix it has “INT” data type.

B. Get the time range between which the orders were placed.

Query:

```
SELECT
  DATE_DIFF(last_order,a.first_order,DAY) AS Range_in_DAYS,
  DATE_DIFF(last_order,a.first_order,MONTH) AS Range_in_MONTH,
  DATE_DIFF(last_order,a.first_order,YEAR) AS Range_in_YEAR,
FROM
(SELECT
  DATE(MIN(order_purchase_timestamp)) AS first_order,
  DATE(MAX(order_purchase_timestamp)) AS last_order
FROM `Target.orders` ) AS a
```

Output screenshot:



Insights:

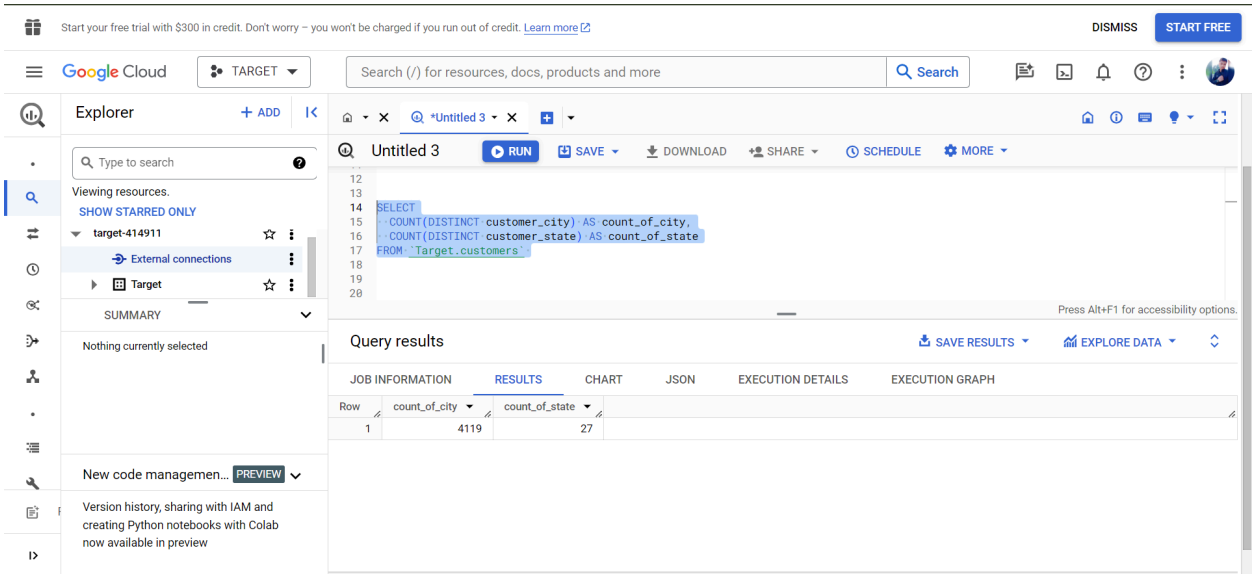
- We have data of 2 years and there are lots of transactions that were placed in the span of 2 years so we have lots of data to analyse.

C. Count the Cities & States of customers who ordered during the given period.

Query:

```
SELECT
  COUNT(DISTINCT customer_city) AS count_of_city,
  COUNT(DISTINCT customer_state) AS count_of_state
FROM `Target.customers`
```

Output screenshot:



Insights:

- There are 4,119 cities and 27 states from where orders are placed.

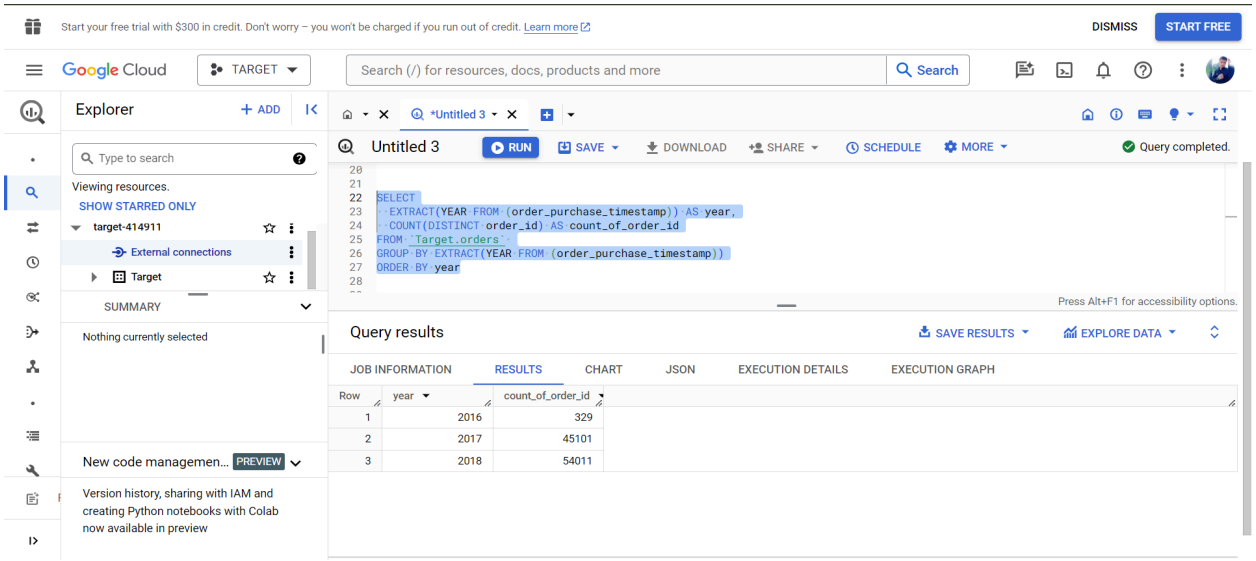
II. In-depth Exploration:

A. Is there a growing trend in the no. of orders placed over the past years?

Query:

```
SELECT
  EXTRACT(YEAR FROM (order_purchase_timestamp)) AS year,
  COUNT(DISTINCT order_id) AS count_of_order_id
FROM `Target.orders`
GROUP BY EXTRACT(YEAR FROM (order_purchase_timestamp))
ORDER BY year
```

Output screenshot:



Insights:

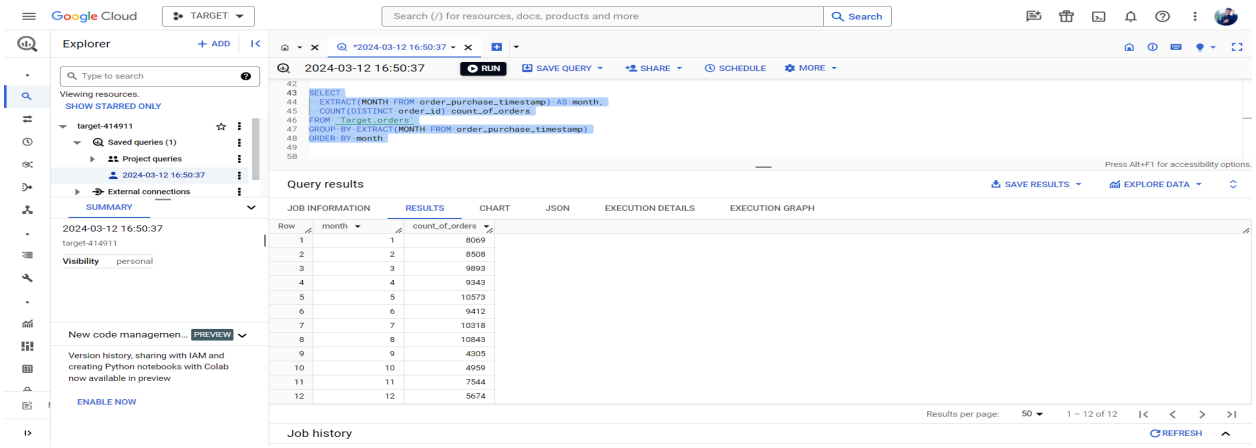
- There is increase in trend
- The order is increasing over the past years
- There is positive business growth

B. Can we see some kind of monthly seasonality in terms of the no. of orders being placed?

Query:

```
SELECT
  EXTRACT(MONTH FROM order_purchase_timestamp) AS month,
  COUNT(DISTINCT order_id) count_of_orders
FROM `Target.orders`
GROUP BY EXTRACT(MONTH FROM order_purchase_timestamp)
ORDER BY month
```

Output screenshot:



Insights:

- In the month of May,July and August there is highest sales.
- In the month of Sep and Oct there is lowest sales

Recommendation :

- In the month of Sep and Oct Company should give offers to their customers so it can increase its sales.

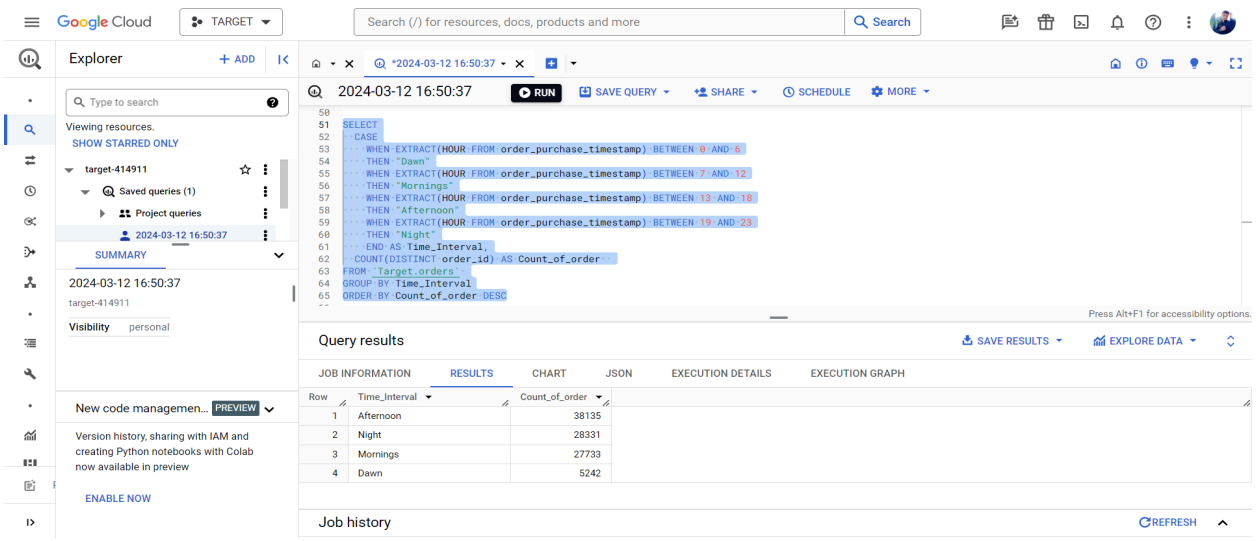
C. During what time of the day, do the Brazilian customers mostly place their orders? (Dawn, Morning, Afternoon or Night)

- **0-6 hrs : Dawn**
- **7-12 hrs : Mornings**
- **13-18 hrs : Afternoon**
- **19-23 hrs : Night**

Query:

```
SELECT
  CASE
    WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 0 AND 6
    THEN "Dawn"
    WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 7 AND 12
    THEN "Mornings"
    WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 13 AND 18
    THEN "Afternoon"
    WHEN EXTRACT(HOUR FROM order_purchase_timestamp) BETWEEN 19 AND 23
    THEN "Night"
    END AS Time_Interval,
  COUNT(DISTINCT order_id) AS Count_of_order
FROM `Target.orders`
GROUP BY Time_Interval
ORDER BY Count_of_order DESC
```

Output screenshot:



Insights:

- **Afternoon is the peak period when Brazilian customers are most active in making purchases.**

- **Afternoon orders may indicate that Brazilian customers prefer to engage in shopping activities during their leisure time, perhaps during breaks from work or other daily responsibilities. This behavior may be influenced by factors such as convenience and availability.**

Recommendation:

- **At Afternoon time company should adjust their operational processes to better accommodate fluctuations in demand. For instance, staffing levels, inventory management, and logistics arrangements can be optimized to ensure efficient order processing and delivery during high-demand periods.**

- **Company can tailor their services based on the identified order placement trends. For example, offering special promotions or customer support services during Afternoon can enhance the overall shopping experience and drive customer satisfaction.**

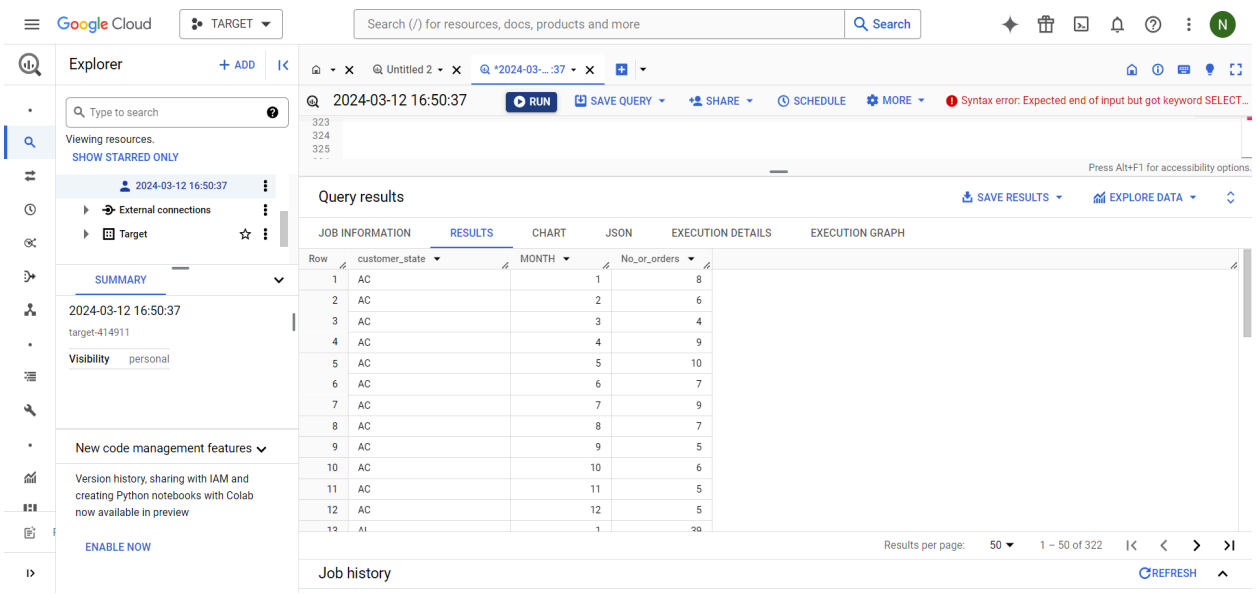
III. Evolution of E-commerce orders in the Brazil region:

A. Get the month on month no. of orders placed in each state.

Query:

```
SELECT
  c.customer_state,
  EXTRACT(MONTH FROM o.order_purchase_timestamp) AS MONTH,
  COUNT(DISTINCT o.order_id) AS No_or_orders
FROM `Target.orders` AS o
LEFT JOIN `Target.payments` AS p
ON o.order_id = p.order_id
LEFT JOIN `Target.customers` AS c
ON o.customer_id = c.customer_id
GROUP BY MONTH,c.customer_state
ORDER BY c.customer_state,MONTH
```

Output screenshot:



Insights:

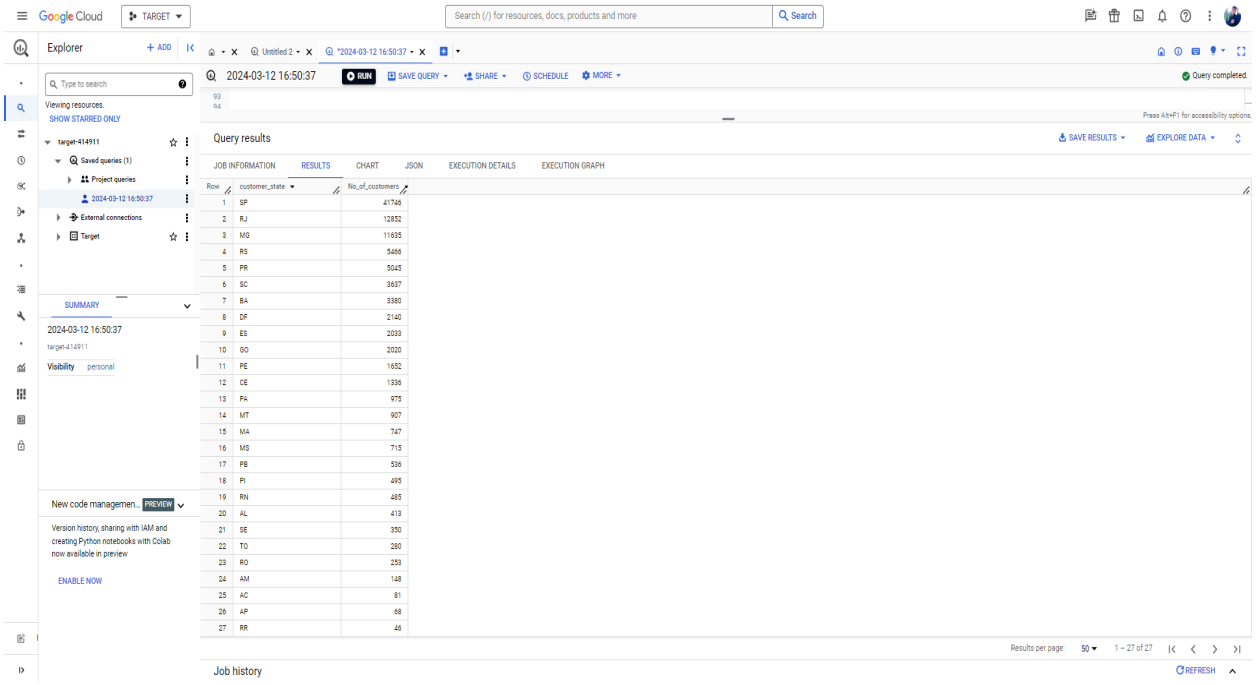
- SP has the lowest sales in the month of “Sep”, "Nov " and "Dec ".
- GO has the lowest sales in “Sep”.
- There is a reduction in sales at “MG” after “Aug”.
- There is a reduction in sales at “RJ” after “Aug”.

B. How are the customers distributed across all the states?

Query:

```
SELECT
  customer_state,
  COUNT(DISTINCT customer_id) AS No_of_customers
FROM `Target.customers`
GROUP BY customer_state
ORDER BY No_of_customers DESC
```

Output screenshot:



Insights:

- **"SP" has the highest number of customers.**
- **"RR" has the lowest number of customers.**

Recommendation:

- **Since "RR" has the lowest number of customers so there is a growth opportunity in this state, the company should focus on "RR" to expand its customer base.**
- **Company should allocate fewer resources as compared to other states.**

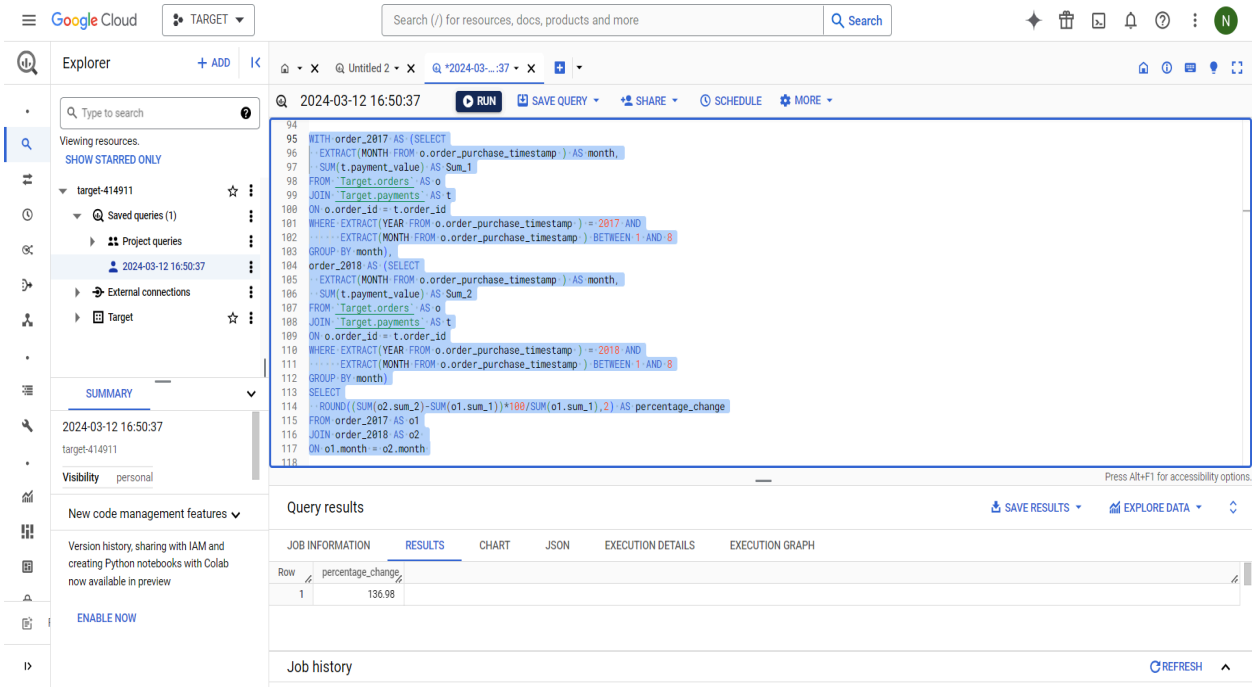
IV. Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.

A. Get the % increase in the cost of orders from year 2017 to 2018 (include months between Jan to Aug only).

Query:

```
WITH order_2017 AS (SELECT
  EXTRACT(MONTH FROM o.order_purchase_timestamp ) AS month,
  SUM(t.payment_value) AS Sum_1
FROM `Target.orders` AS o
JOIN `Target.payments` AS t
ON o.order_id = t.order_id
WHERE EXTRACT(YEAR FROM o.order_purchase_timestamp ) = 2017 AND
      EXTRACT(MONTH FROM o.order_purchase_timestamp ) BETWEEN 1 AND 8
GROUP BY month),
order_2018 AS (SELECT
  EXTRACT(MONTH FROM o.order_purchase_timestamp ) AS month,
  SUM(t.payment_value) AS Sum_2
FROM `Target.orders` AS o
JOIN `Target.payments` AS t
ON o.order_id = t.order_id
WHERE EXTRACT(YEAR FROM o.order_purchase_timestamp ) = 2018 AND
      EXTRACT(MONTH FROM o.order_purchase_timestamp ) BETWEEN 1 AND 8
GROUP BY month)
SELECT
  ROUND((SUM(o2.sum_2)-SUM(o1.sum_1))*100/SUM(o1.sum_1),2) AS percentage_change
FROM order_2017 AS o1
JOIN order_2018 AS o2
ON o1.month = o2.month
```

Output screenshot:



Insights:

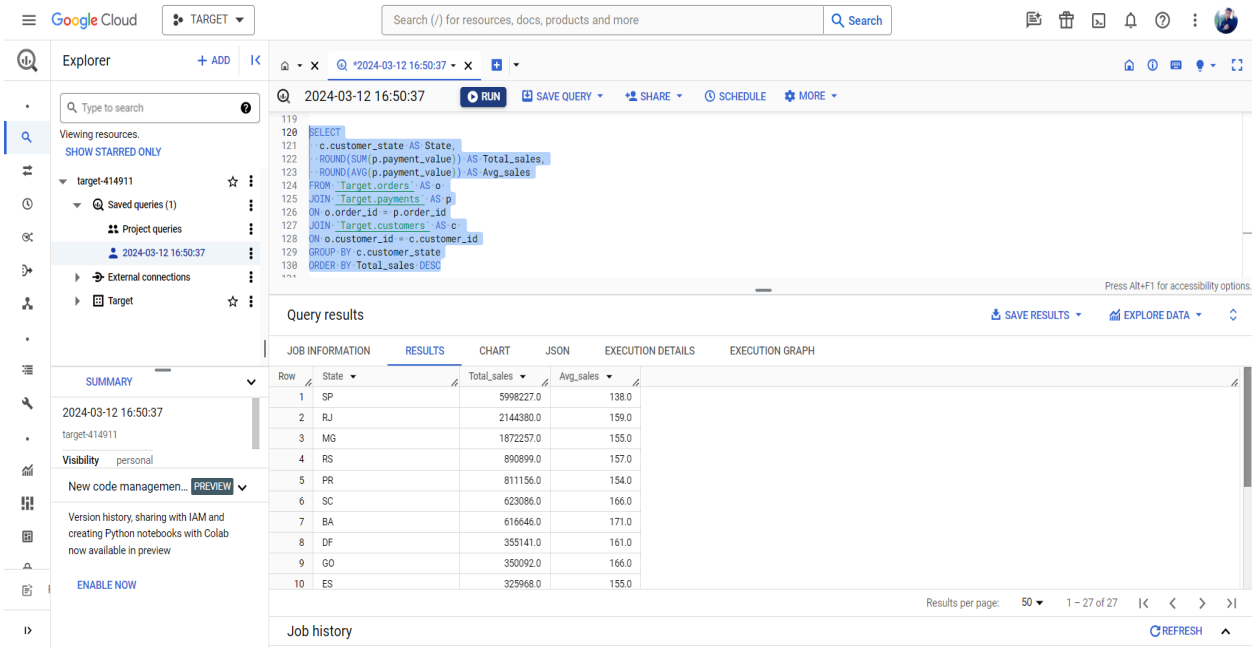
- There is a 136.98% increase in sales from 2017 to 2018.
- We have taken the sales data only from Jan to Aug for both the years.
- There is an increasing trend.

B. Calculate the Total & Average value of order price for each state.

Query:

```
SELECT
  c.customer_state AS State,
  ROUND(SUM(p.payment_value)) AS Total_sales,
  ROUND(AVG(p.payment_value)) AS Avg_sales
FROM `Target.orders` AS o
JOIN `Target.payments` AS p
ON o.order_id = p.order_id
JOIN `Target.customers` AS c
ON o.customer_id = c.customer_id
GROUP BY c.customer_state
ORDER BY Total_sales DESC
```

Output screenshot:



Insights:

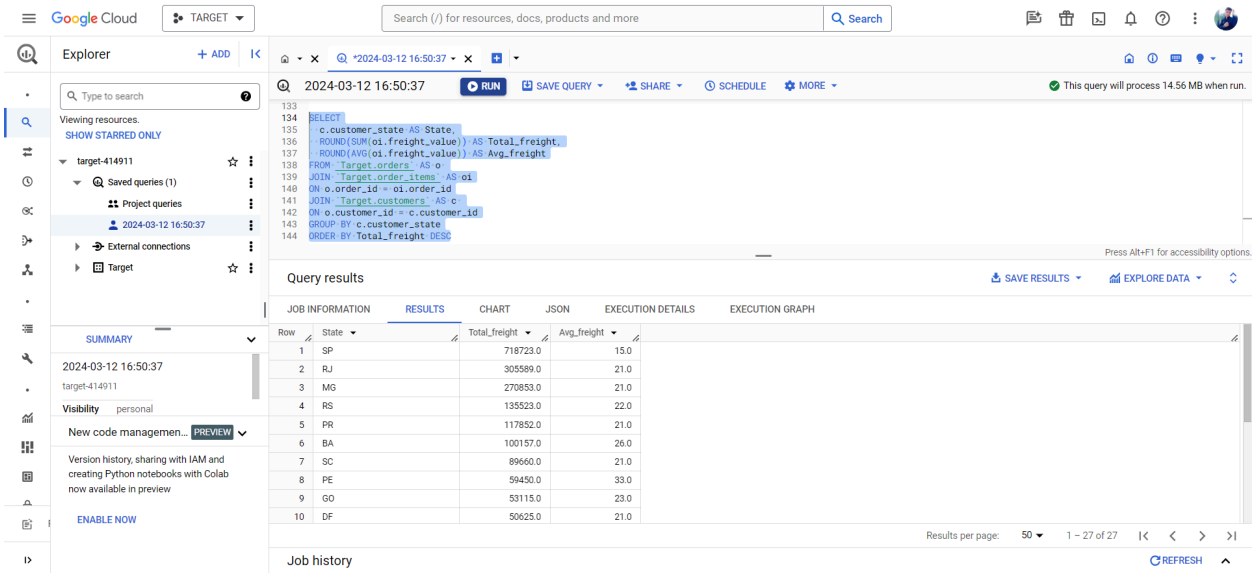
- **"SP" has the highest total sales but lowest average sales. It means the number of customers are higher but they spend a low amount on their purchase.**
- **"RR" has the lowest total sales.**
- **"PB" has the highest average sales.**

C. Calculate the Total & Average value of order freight for each state.

Query:

```
SELECT
  c.customer_state AS State,
  ROUND(SUM(oi.freight_value)) AS Total_freight,
  ROUND(AVG(oi.freight_value)) AS Avg_freight
FROM `Target.orders` AS o
JOIN `Target.order_items` AS oi
ON o.order_id = oi.order_id
JOIN `Target.customers` AS c
ON o.customer_id = c.customer_id
GROUP BY c.customer_state
ORDER BY Total_freight DESC
```

Output screenshot:



Insights:

- "SP" has the highest total freight and lowest average freight
- "RR" has the lowest total freight.
- "PB" has the highest average freight.

V. Analysis based on sales, freight and delivery time.

A. Find the no. of days taken to deliver each order from the order's purchase date as delivery time.

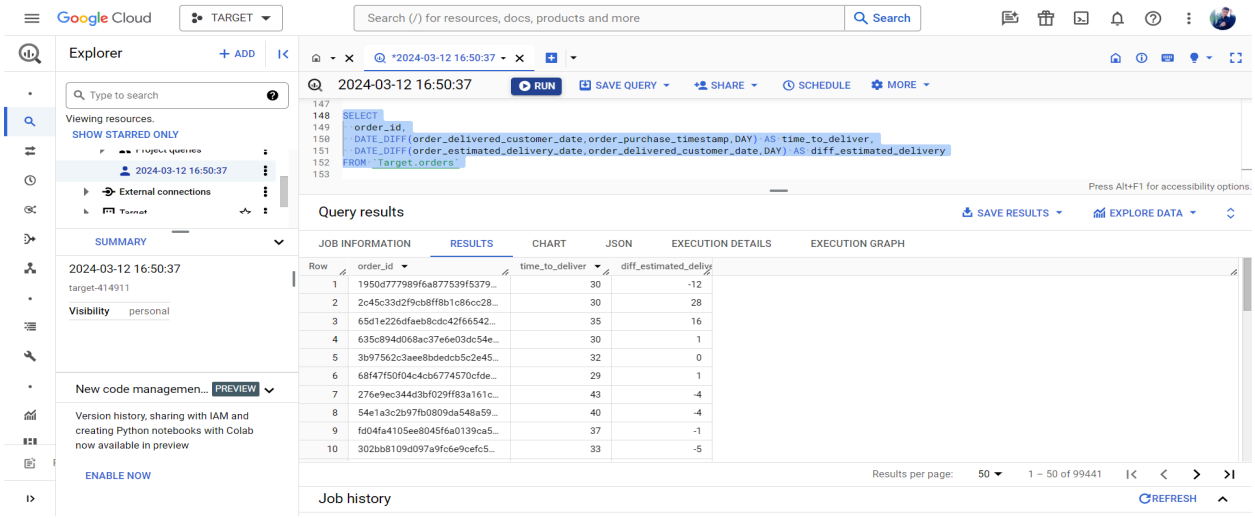
Also, calculate the difference (in days) between the estimated & actual delivery date of an order.

Do this in a single query.

Query:

```
SELECT
  order_id,
       DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY)      AS
time_to_deliver,
       DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY)   AS
diff_estimated_delivery
FROM `Target.orders`
```

Output screenshot:



Insights:

- **Maximum delivery time is 209 days.**
- **Minimum delivery day is the same day.**
- **Maximum delay in delivery over estimation is 188 days. It shows the inefficient logistics.**

- **Since there is too much of a gap between estimated date and actual delivery date, it means it does not give estimated date accurately and so on customers will not rely on their estimated delivery date.**

Recommendation:

- **If there are significant delays between the estimated and actual delivery dates for certain orders, it may indicate potential bottlenecks in the company's supply chain or logistics operations. Identifying these bottlenecks can help company to streamline processes and improve overall efficiency.**

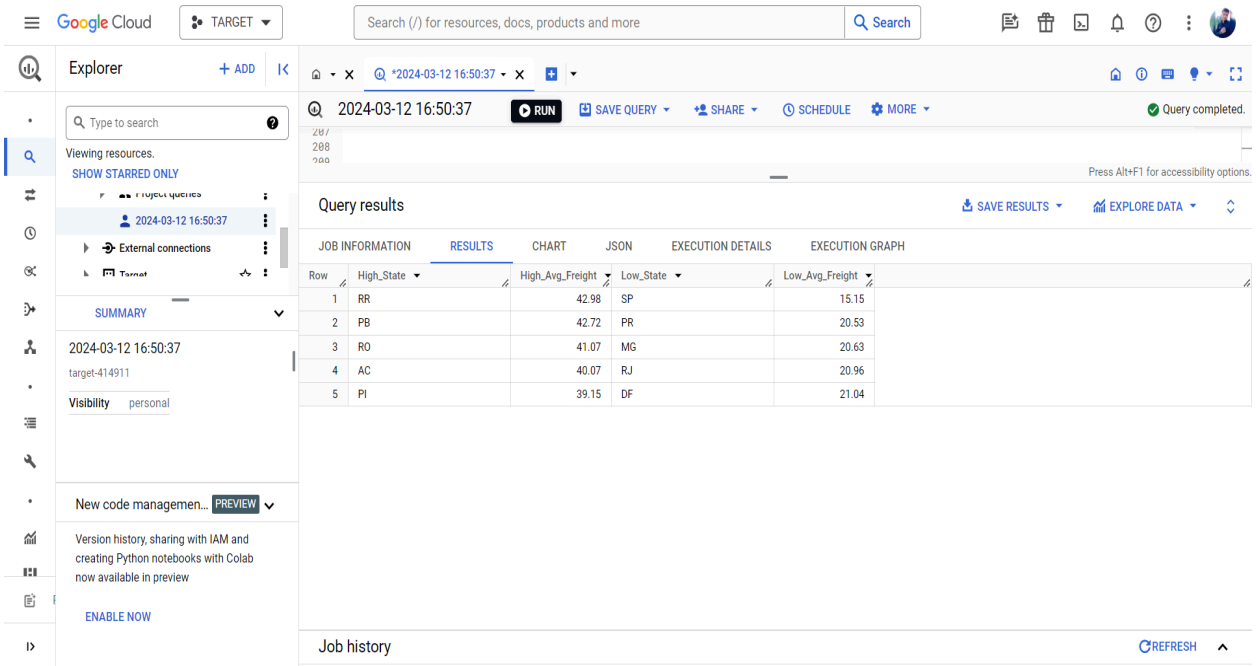
B. Find out the top 5 states with the highest & lowest average freight value.

Query:

```
SELECT
  x.State AS High_State,
  x.Avg_freight AS High_Avg_Freight,
  y.State AS Low_State,
  y.Avg_freight AS Low_Avg_Freight
FROM
(SELECT
  *,
  DENSE_RANK() OVER(ORDER BY Avg_freight DESC) AS rank_
FROM
(SELECT
  c.customer_state AS State,
  ROUND(AVG(oi.freight_value),2) AS Avg_freight,
FROM `Target.orders` AS o
JOIN `Target.order_items` AS oi
ON o.order_id = oi.order_id
JOIN `Target.customers` AS c
ON o.customer_id = c.customer_id
GROUP BY c.customer_state
ORDER BY Avg_freight DESC) AS n
ORDER BY rank_
LIMIT 5) AS x
```

```
JOIN
(SELECT
  *,
  DENSE_RANK() OVER(ORDER BY Avg_freight ASC) AS rank_
FROM
(SELECT
  c.customer_state AS State,
  ROUND(AVG(oi.freight_value),2) AS Avg_freight,
FROM `Target.orders` AS o
JOIN `Target.order_items` AS oi
ON o.order_id = oi.order_id
JOIN `Target.customers` AS c
ON o.customer_id = c.customer_id
GROUP BY c.customer_state
ORDER BY Avg_freight ASC) AS n
ORDER BY rank_
LIMIT 5) AS y
ON x.rank_ = y.rank_
```

Output screenshot:



Insights:

- **"RR", "PB", "RO", "AC" and "PI" are states with higher average freight. In these states shipping costs are generally higher. These States may face challenges in terms of accessibility or transportation infrastructure, leading to higher shipping costs.**

- **"SP", "PR", "MG", "RJ" and "DF" are states with lower average freight. In these states shipping costs are generally lower. These states may have better access to transportation routes or logistics hubs.**

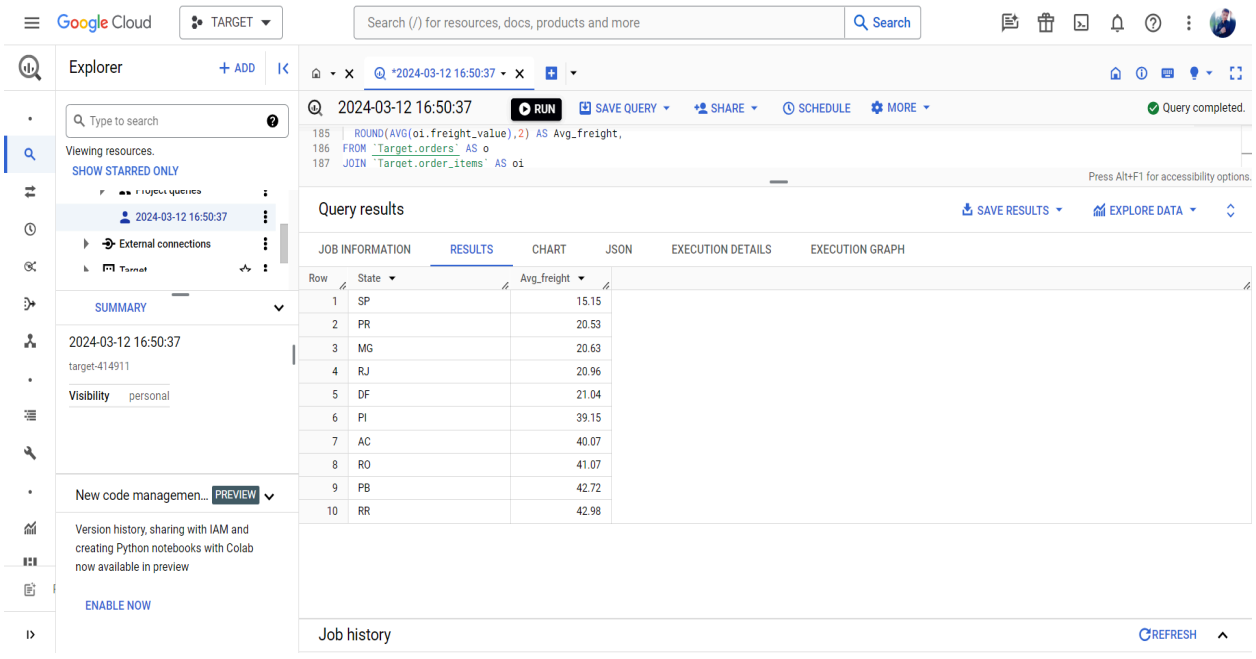
C. Find out the top 5 states with the highest & lowest average delivery time.

Query:

```
SELECT
  State,
  Avg_freight
FROM
(SELECT
  *
FROM
(SELECT
  *,
  DENSE_RANK() OVER(ORDER BY Avg_freight DESC) AS rank_
FROM
(SELECT
  c.customer_state AS State,
  ROUND(AVG(oi.freight_value),2) AS Avg_freight,
FROM `Target.orders` AS o
JOIN `Target.order_items` AS oi
ON o.order_id = oi.order_id
JOIN `Target.customers` AS c
ON o.customer_id = c.customer_id
GROUP BY c.customer_state
ORDER BY Avg_freight DESC) AS n
```

```
ORDER BY rank_
LIMIT 5)
UNION ALL
(SELECT
  *,
  DENSE_RANK() OVER(ORDER BY Avg_freight ASC) AS rank_
FROM
(SELECT
  c.customer_state AS State,
  ROUND(AVG(oi.freight_value),2) AS Avg_freight,
FROM `Target.orders` AS o
JOIN `Target.order_items` AS oi
ON o.order_id = oi.order_id
JOIN `Target.customers` AS c
ON o.customer_id = c.customer_id
GROUP BY c.customer_state
ORDER BY Avg_freight ASC) AS m
ORDER BY rank_
LIMIT 5)) AS k
ORDER BY k.Avg_freight
```

Output screenshot:



Insights:

- **"RR", "PB", "RO","AC" and "PI" are states with higher average delivery time. Analysing average delivery times across different states can help identify areas where customers may be less satisfied due to longer delivery times. Addressing these issues could lead to improved customer retention and loyalty.**

- **"SP", "PR", "MG", "RJ" and "DF" are states with lowest average delivery time. Identifying the factors contributing to these shorter delivery times could provide insights into best practices that could be applied to other regions to improve overall efficiency.**

D. Find out the top 5 states where the order delivery is really fast as compared to the estimated date of delivery.

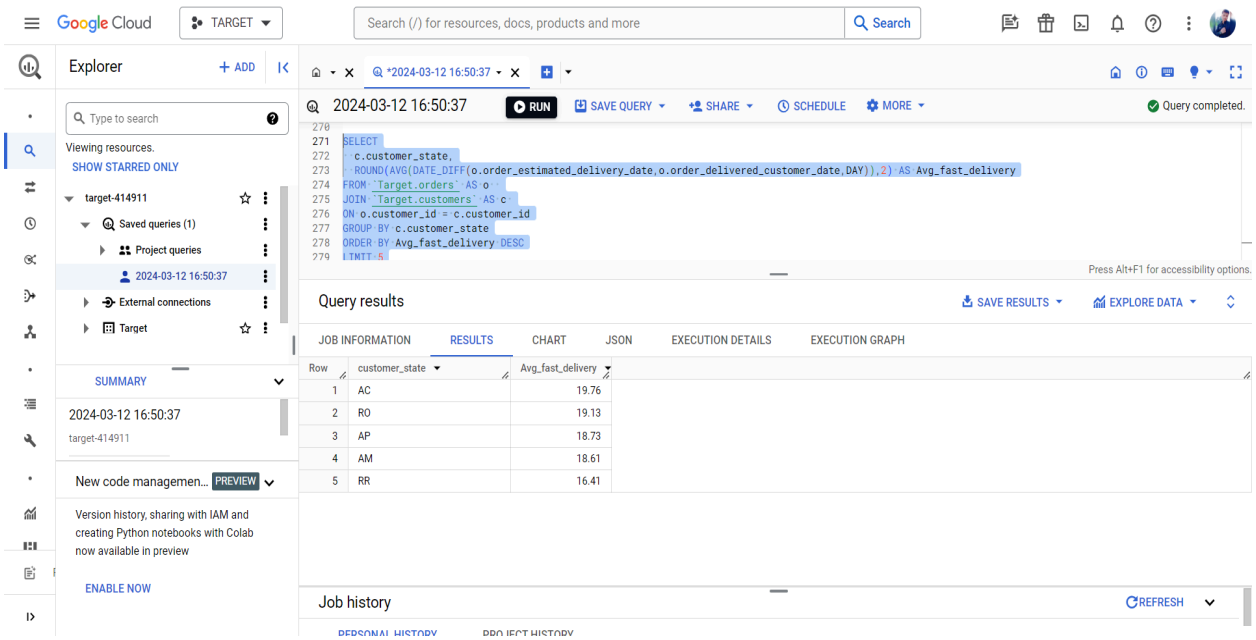
You can use the difference between the averages of actual & estimated delivery date to figure out how fast the delivery was for each state.

Query:

```
SELECT
  c.customer_state,

ROUND(AVG(DATE_DIFF(o.order_estimated_delivery_date,o.order_delivered_customer_date,DA
Y)),2) AS Avg_fast_delivery
FROM `Target.orders` AS o
JOIN `Target.customers` AS c
ON o.customer_id = c.customer_id
GROUP BY c.customer_state
ORDER BY Avg_fast_delivery DESC
LIMIT 5
```

Output screenshot:



Insights:

- **"AC", "RO", "AP", "AM" and "RR" are the states where the order delivery is really fast as compared to the estimated date of delivery.**

- **These are the states where the actual delivery time is consistently faster than the estimated time indicating efficient logistical operations or better-than-expected delivery performance.**

- **The query provides a benchmark for comparing delivery performance across different states. This allows businesses to set targets and track improvements over time, fostering continuous optimization of delivery operations.**

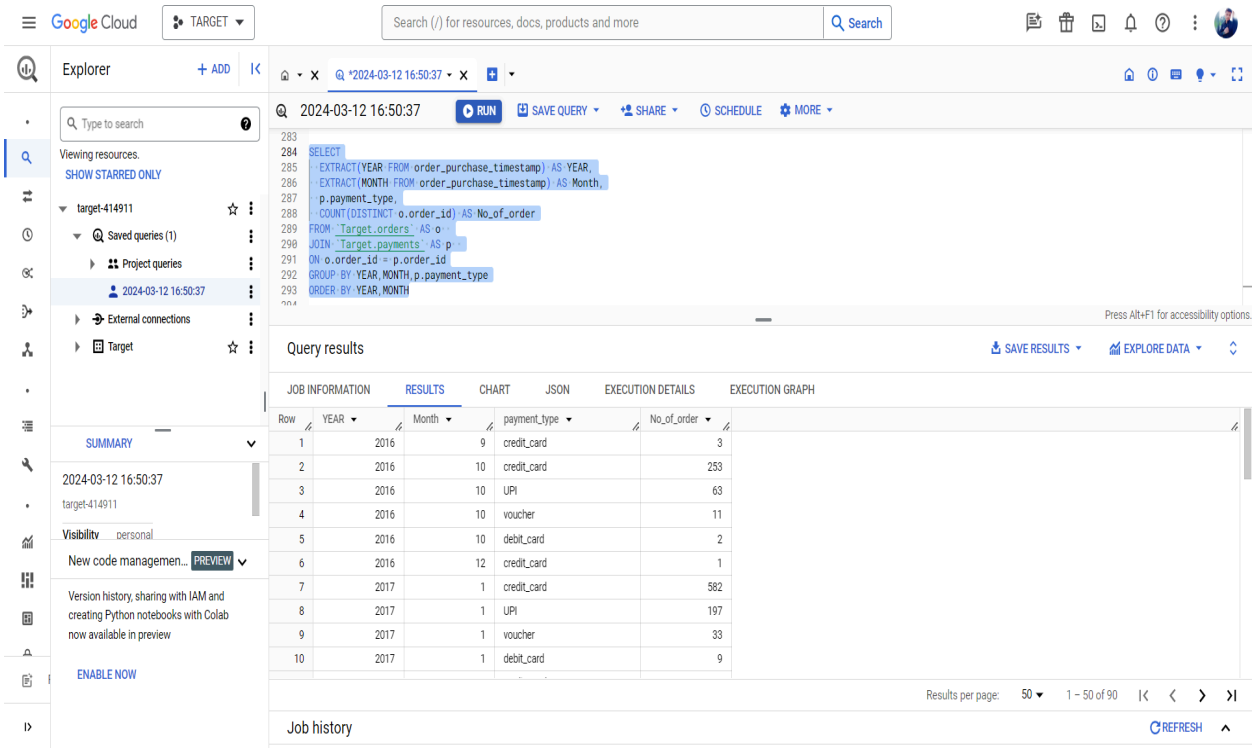
VI. Analysis based on the payments:

A. Find the month on month no. of orders placed using different payment types.

Query:

```
SELECT
  EXTRACT(YEAR FROM order_purchase_timestamp) AS YEAR,
  EXTRACT(MONTH FROM order_purchase_timestamp) AS Month,
  p.payment_type,
  COUNT(DISTINCT o.order_id) AS No_of_order
FROM `Target.orders` AS o
JOIN `Target.payments` AS p
ON o.order_id = p.order_id
GROUP BY YEAR,MONTH,p.payment_type
ORDER BY YEAR,MONTH
```

Output screenshot:



Insights:

- Mostly customers are using "Debit card" and "Credit card".

Recommendation:

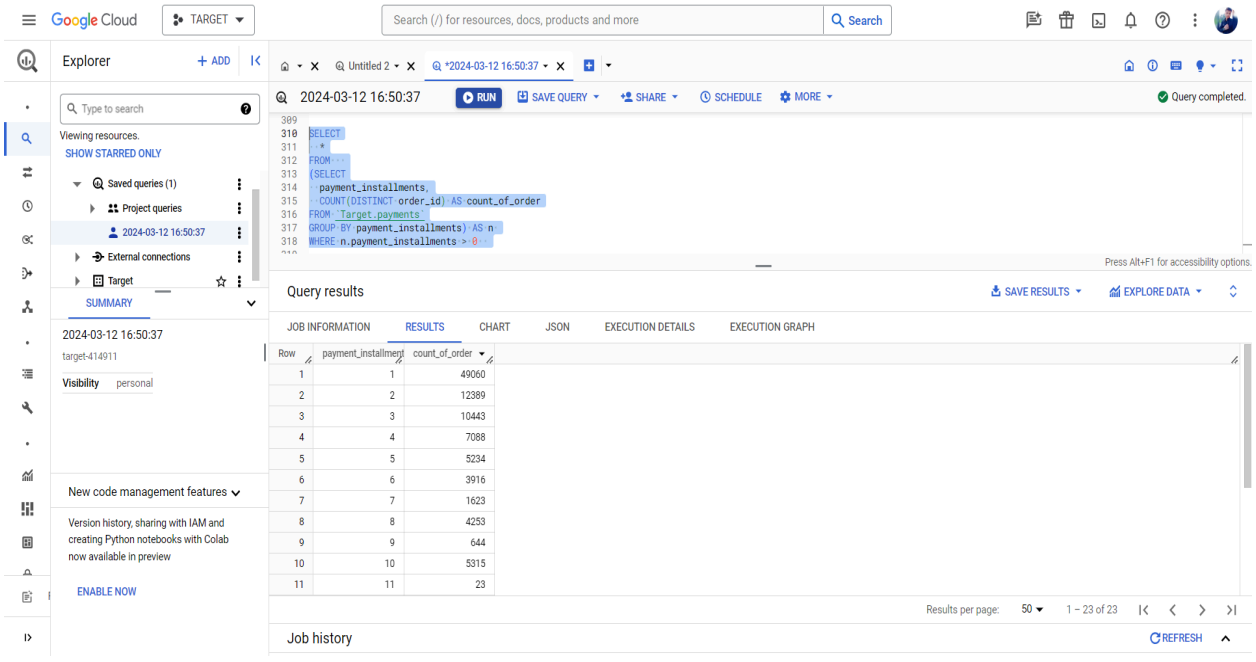
- Company should give purchase through "Debit card" and "Credit card" options on all the products.

B. Find the no. of orders placed on the basis of the payment installments that have been paid.

Query:

```
SELECT
  *
FROM
(SELECT
  payment_installments,
  COUNT(DISTINCT order_id) AS count_of_order
FROM `Target.payments`
GROUP BY payment_installments) AS n
WHERE n.payment_installments > 0
```

Output screenshot:



Insights:

- **There are 49,060 orders who have paid their 1st installments but when the number of installments are increasing the number of orders who are paying there installments are decreasing.**