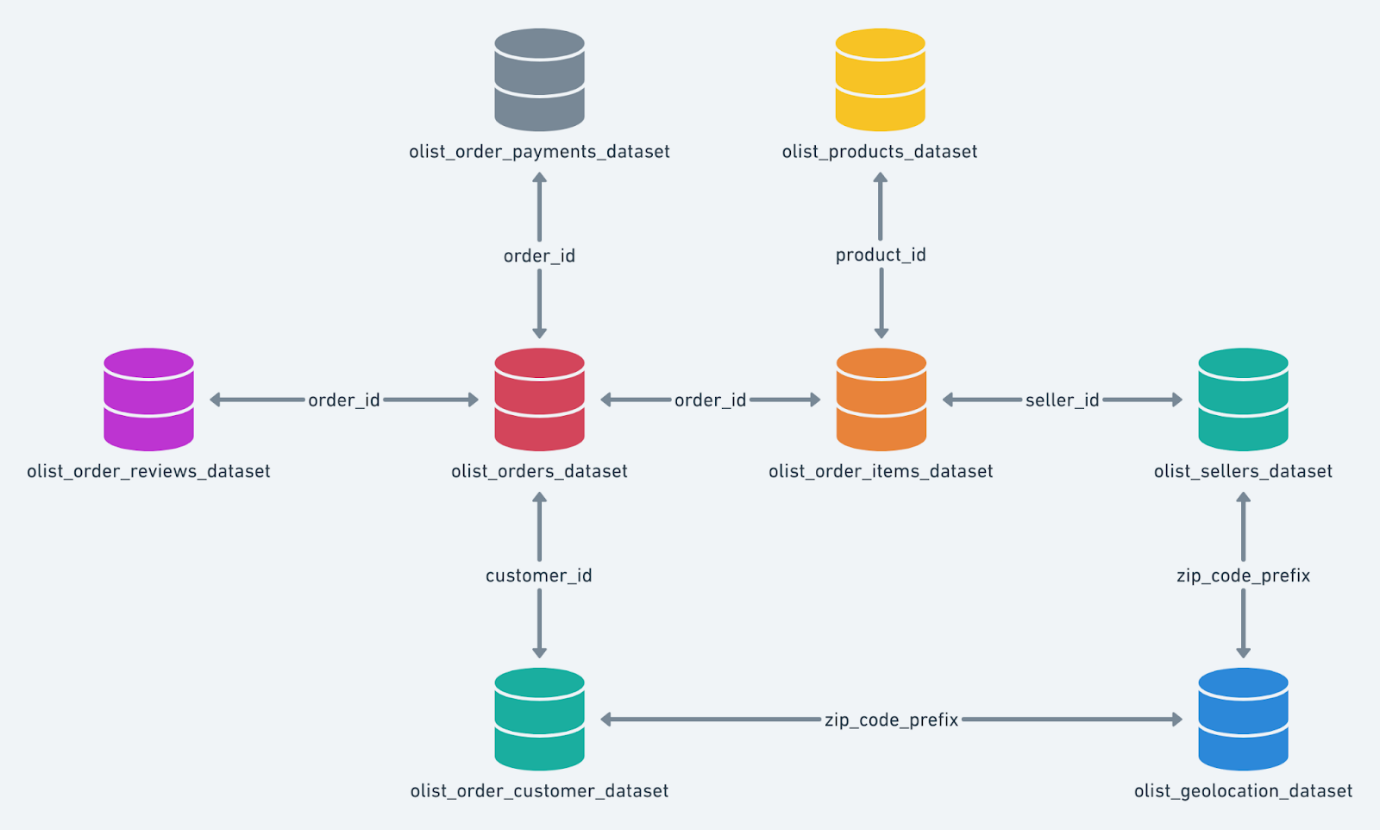
**Business Case: Target SQL**

Data from 100,000 orders placed at Target in Brazil between 2016 and 2018 is included in this business case. It is the top retail chain in America.   
Eight tables include data about orders from several aspects, including order status, payment information, order location and time, client who made the purchase, items in the order, product details, vendor information, order reviews, etc.



**ANAYLISIS**

1. **Initial exploration of dataset**
   * 1. Data type of all columns of different tables in the "Target" dataset.

**🡪 🡪** *Query:*

SELECT

  table\_schema,

  table\_name,

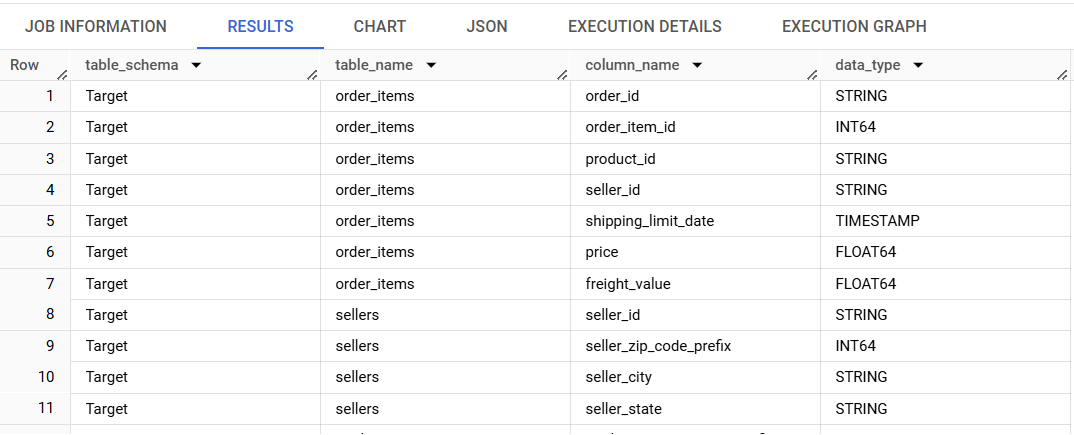
  column\_name,

  data\_type

FROM

  `Target`.INFORMATION\_SCHEMA.COLUMNS;

**🡪 🡪** *Result*:



* + 1. The timespan of available data.

**🡪 🡪** *Query:*

SELECT

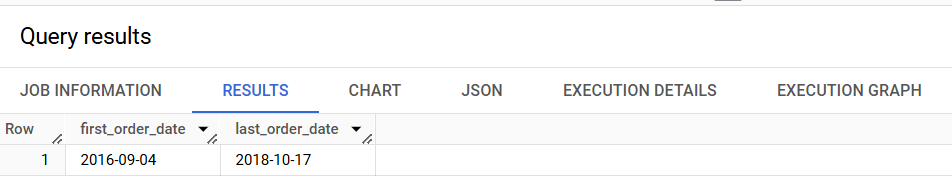
  MIN(DATE(order\_purchase\_timestamp)) AS first\_order\_date,

  MAX(DATE(order\_purchase\_timestamp)) AS last\_order\_date

FROM

  `Target.orders`;

**🡪 🡪** *Result*:



* + 1. Number of cities per State of customers who ordered during the given period.

**🡪 🡪** *Query:*

SELECT

  DISTINCT c.customer\_state , count (c.customer\_city) as No\_of\_Cities

FROM

  `Target.customers` c

RIGHT JOIN

  `Target.orders` o

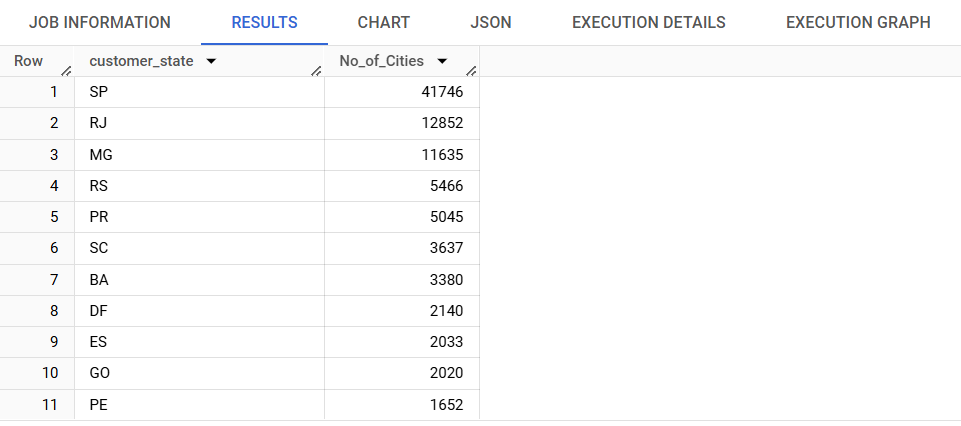
ON

  c.customer\_id = o.customer\_id

GROUP BY c.customer\_state

ORDER BY No\_of\_Cities DESC, c.customer\_state;

**🡪 🡪** *Result*:



* + 1. Distribution of total orders as per their status.

**🡪 🡪** *Query:*

SELECT

  DISTINCT c.customer\_state , count (c.customer\_city) as No\_of\_Cities

FROM

  `Target.customers` c

RIGHT JOIN

  `Target.orders` o

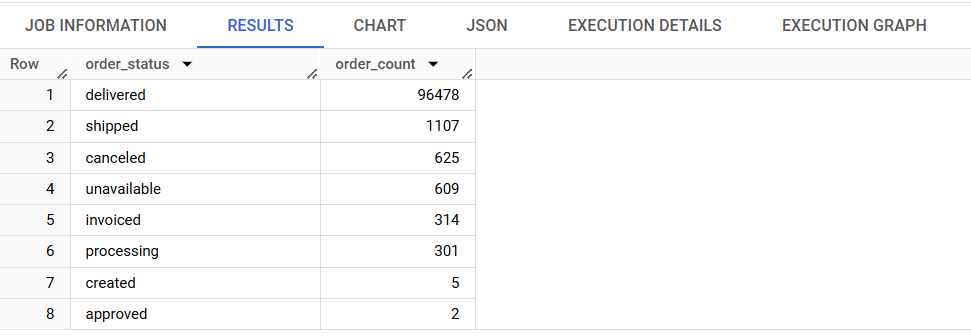
ON

  c.customer\_id = o.customer\_id

GROUP BY c.customer\_state

ORDER BY No\_of\_Cities DESC, c.customer\_state;

**🡪 🡪** *Result*:



1. **In-Depth Analysis**
   * 1. Top 10 cities with largest customer base.

**🡪 🡪** *Query:*

SELECT

time\_period, order\_count,

ROUND((((order\_count - LAG(order\_count) OVER(ORDER BY year, month)) / LAG(order\_count) OVER(ORDER BY year, month))\* 100), 2) AS growth\_percent

FROM

(SELECT

EXTRACT (MONTH FROM order\_purchase\_timestamp) AS month,

EXTRACT (YEAR FROM order\_purchase\_timestamp) AS year,

FORMAT\_DATE('%b %Y', DATE(order\_purchase\_timestamp)) AS time\_period,COUNT(order\_id) AS order\_count

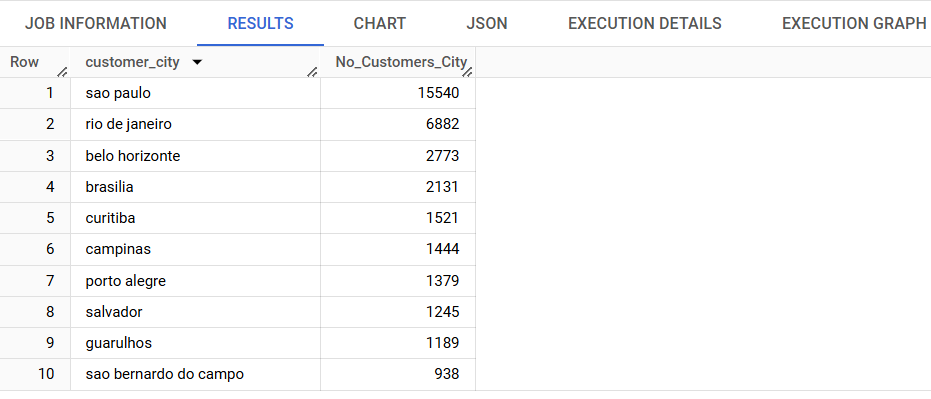
FROM `Target.orders`

WHERE order\_status = 'delivered'

GROUP BY month, year, time\_period) as T1

ORDER BY year, month;

**🡪 🡪** *Result*:



* + 1. Top 10 cities with the greatest number of orders.

**🡪 🡪** *Query:*

SELECT

  Tempo.geolocation\_city, COUNT(Tempo.geolocation\_city) as Orders\_per\_City

FROM

  `Target.orders` as o

INNER JOIN

  (SELECT c.customer\_id ,g.geolocation\_city

  FROM

    `Target.customers` as c

  INNER JOIN

    `Target.geolocation` as g

  ON

    c.customer\_zip\_code\_prefix = g.geolocation\_zip\_code\_prefix) AS Tempo

ON

  Tempo.customer\_id = o.customer\_id

GROUP BY

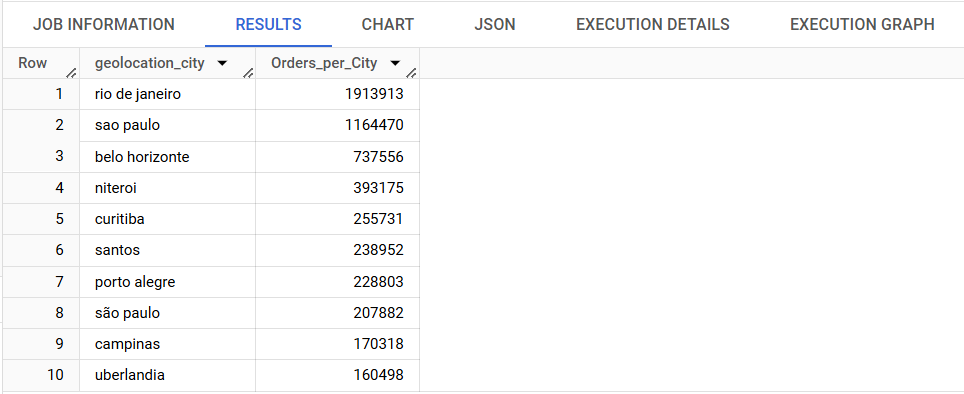
  Tempo.geolocation\_city

ORDER BY

  Orders\_per\_City DESC

LIMIT 10

**🡪 🡪** *Result*:



* + 1. Is there a growing trend in the no. of orders placed over the past years?

**🡪 🡪** *Query:*

SELECT

time\_period, order\_count,

ROUND((((order\_count - LAG(order\_count) OVER(ORDER BY year, month)) / LAG(order\_count) OVER(ORDER BY year, month))\* 100), 2) AS growth\_percent

FROM

(SELECT

EXTRACT (MONTH FROM order\_purchase\_timestamp) AS month,

EXTRACT (YEAR FROM order\_purchase\_timestamp) AS year,

FORMAT\_DATE('%b %Y', DATE(order\_purchase\_timestamp)) AS time\_period,COUNT(order\_id) AS order\_count

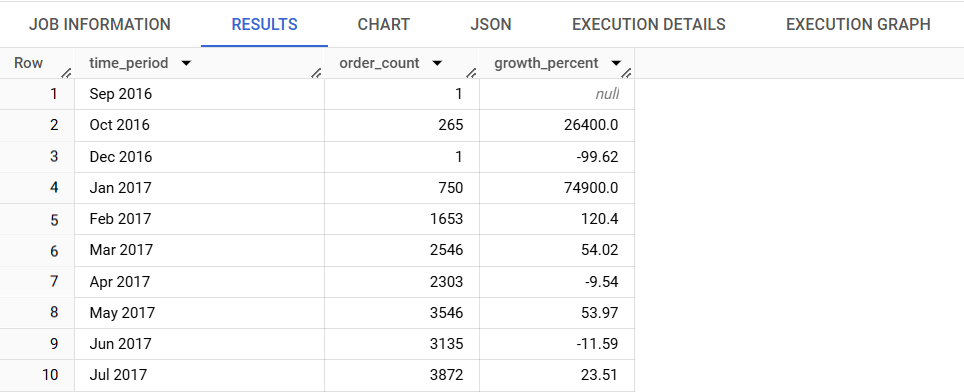
FROM `Target.orders`

WHERE order\_status = 'delivered'

GROUP BY month, year, time\_period) as T1

ORDER BY year, month;

**🡪 🡪** *Result*:

****

There is no concrete evidence to show any pattern that shows seasonality in orders placed. However, it is seen that there is growing trend in no. of orders placed over the past years.

* + 1. What time do Brazilian customers tend to buy (Dawn, Morning, Afternoon or Night)?

**🡪 🡪** *Query:*

SELECT order\_time, count(\*) AS Total\_Orders

FROM

(SELECT

    order\_id,

    CASE

      WHEN TIME(order\_purchase\_timestamp) BETWEEN "00:00:00" AND "07:00:00" THEN "Dawn"

      WHEN TIME(order\_purchase\_timestamp) BETWEEN "07:00:01" AND "12:00:00" THEN "Morning"

      WHEN TIME(order\_purchase\_timestamp) BETWEEN "12:00:01" AND "18:00:00" THEN "Afternoon"

      WHEN TIME(order\_purchase\_timestamp) BETWEEN "18:00:01" AND "23:59:59" THEN "Night"

    END AS order\_time

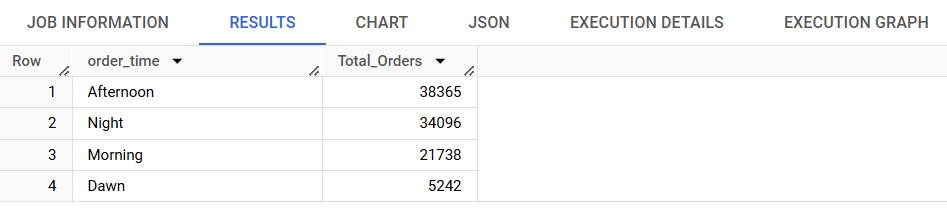
  FROM

    `Target.orders`) AS ORDER\_SLOTS

GROUP BY order\_time

ORDER BY Total\_Orders DESC

**🡪 🡪** *Result*:



It is seen that the Brazilian customers tend to place most of their orders during afternoon and night.

1. **Evolution of E-commerce orders in the Brazil region**
   * 1. Month on month orders by states

**🡪 🡪** *Query:*

SELECT

  state ,year, month, time\_period , total\_orders,

  LAG(total\_orders) OVER(PARTITION BY state ORDER BY year, month ) AS prev\_month\_orders\_count,

  ROUND(((total\_orders - LAG(total\_orders) OVER(PARTITION BY state ORDER BY year, month )) / LAG(total\_orders) OVER(PARTITION BY state ORDER BY year, month))\* 100,2) AS MoM\_percent\_growth

FROM (

  SELECT

    state, time\_period, year,month,

    COUNT(\*) AS total\_orders

  FROM (

    SELECT

      o.order\_id, o.order\_purchase\_timestamp,

      EXTRACT(YEAR FROM order\_purchase\_timestamp) AS year,

      EXTRACT(Month FROM order\_purchase\_timestamp) AS month,

      FORMAT\_DATE('%b %Y', DATE(order\_purchase\_timestamp)) AS time\_period,

      c.customer\_state AS state

    FROM

      `Target.orders` o

    JOIN

      `Target.customers` c

    USING

      (customer\_id)

    ORDER BY

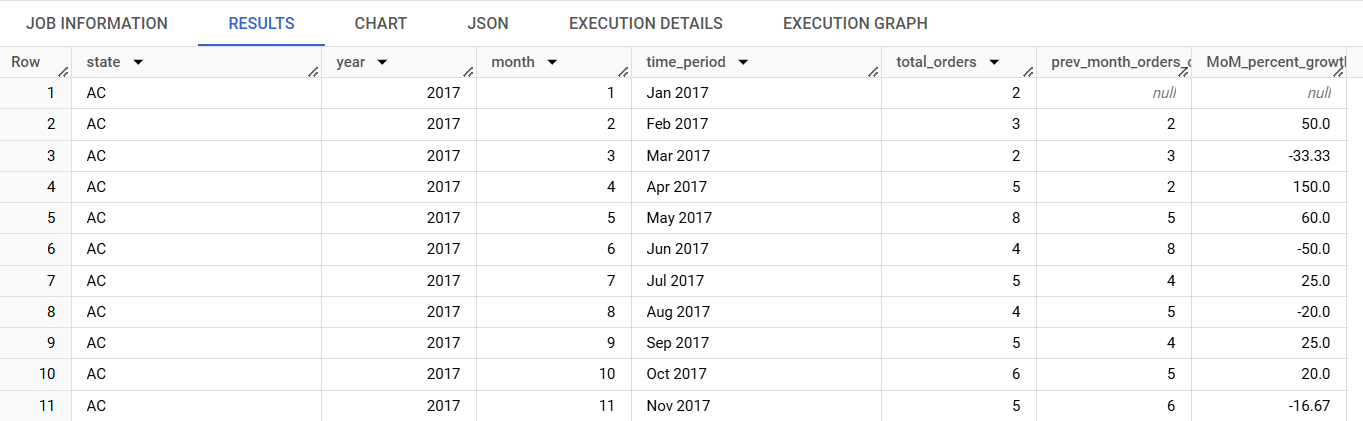
       year, month) Tempo

  GROUP BY

    state, time\_period,year, month) Tempo1

  ORDER BY state, year, month;

**🡪 🡪** *Result*:



* + 1. Distribution of customers across states in Brazil

**🡪 🡪** *Query:*

SELECT

  customer\_state AS State,

  COUNT(\*) AS Total\_Customers

FROM

  `Target.customers`

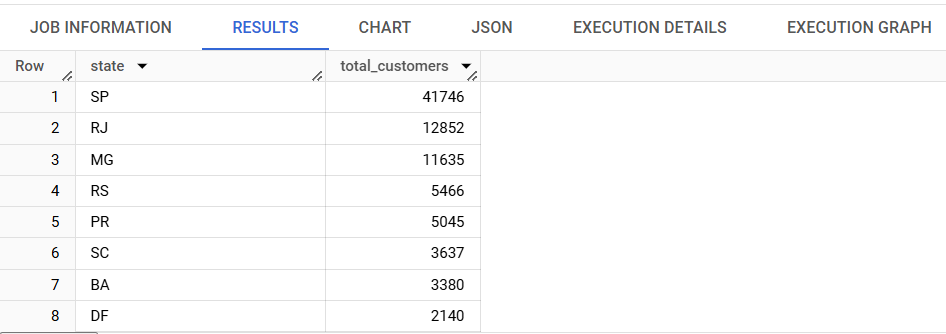
GROUP BY

  customer\_state

ORDER BY

  total\_customers DESC;

**🡪 🡪** *Result*:



1. **Impact on Economy: Analyzing the money movement by e-commerce by looking at order prices, freight and others.**
2. Percentage increase in the cost of orders from year 2017 to 2018 (including months between Jan to Aug only).

**🡪 🡪** *Query:*

SELECT

  \*,

COALESCE((ROUND(((Total\_Order\_Value - LAG(Total\_Order\_Value) OVER(ORDER BY year))/LAG(Total\_Order\_Value) OVER(ORDER BY year))\* 100, 2)), 0) AS YoY

FROM (

  SELECT

    year,

    ROUND(SUM(payment\_value), 2) AS Total\_Order\_Value

  FROM (

    SELECT

      o.order\_id,

      o.order\_purchase\_timestamp,

      EXTRACT(MONTH FROM o.order\_purchase\_timestamp) AS month,

      EXTRACT(YEAR FROM  o.order\_purchase\_timestamp) AS year,

      p.payment\_value

    FROM

      `Target.orders` o

    INNER JOIN

      `Target.payments` p

    USING

      (order\_id)

    WHERE

      o.order\_status = "delivered") AS T

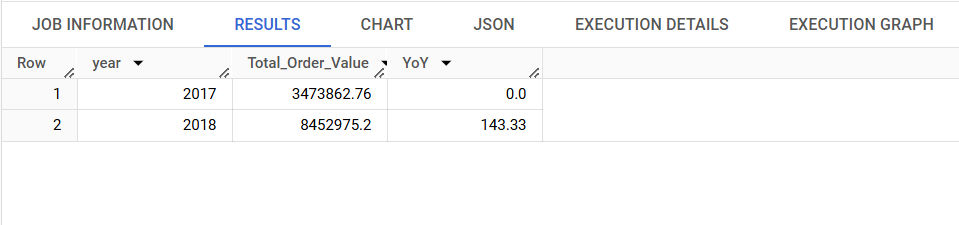
  WHERE

    month BETWEEN 1 AND 8

  GROUP BY year) AS T1

ORDER BY Year;

**🡪 🡪** *Result*:



1. Total & Average value of order price and freight value for each state.

**🡪 🡪** *Query:*

SELECT

  c.customer\_state,

  ROUND(SUM(oi.price)) AS Total\_Price,

  ROUND(AVG(oi.price)) AS Avg\_Price,

  ROUND(SUM(oi.freight\_value)) AS Total\_Freight,

  ROUND(AVG(oi.freight\_value)) AS Avg\_Freight

FROM

  `Target.order\_items` AS oi

INNER JOIN

  `Target.orders` o

ON

  oi.order\_id = o.order\_id

INNER JOIN

  `Target.customers` c

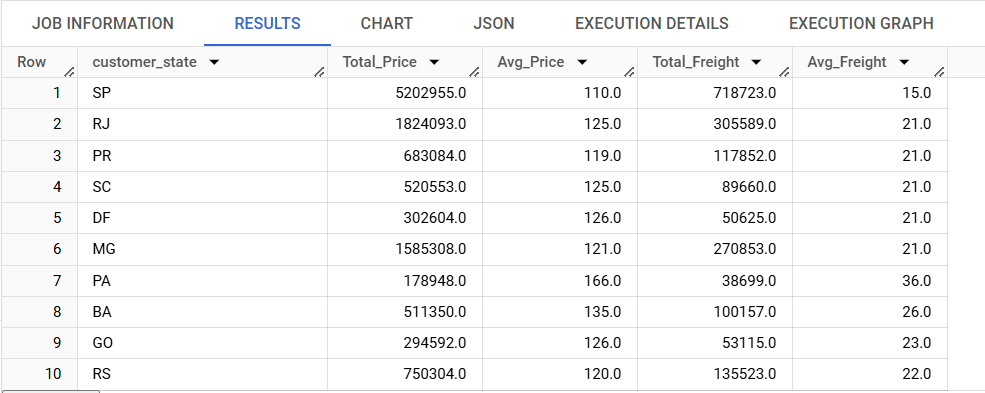
ON

  c.customer\_id = o.customer\_id

GROUP BY

  c.customer\_state;

**🡪 🡪** *Result*:



1. **Analysis based on sales, freight and delivery time.**
2. Number of days taken to deliver each order from the order’s purchase date as delivery time. Along with, the difference (in days) between the estimated & actual delivery date of an order.

**🡪 🡪** *Query:*

SELECT

    \* ,T.Actual\_Delivery\_Time\_in\_Days - T.Estimated\_Delivery\_Time\_in\_Days AS Diff\_Estimated\_Delivery

FROM

(SELECT

  order\_id,

  TIMESTAMP\_DIFF(order\_delivered\_customer\_date, order\_purchase\_timestamp, day) AS Actual\_Delivery\_Time\_in\_Days ,

  TIMESTAMP\_DIFF(order\_estimated\_delivery\_date, order\_purchase\_timestamp, day) AS Estimated\_Delivery\_Time\_in\_Days,

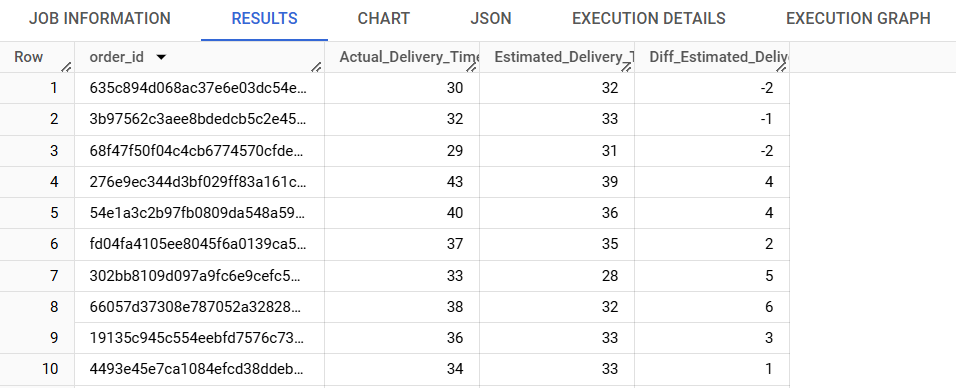
FROM

  `Target.orders`

WHERE

  order\_status = "delivered") AS T;

**🡪 🡪** *Result*:



1. Ranking states with the highest & lowest average freight value.

**🡪 🡪** *Query:*

SELECT

  State,Avg\_Freight\_Value,

  ROW\_NUMBER() OVER (ORDER BY Avg\_Freight\_Value) AS Ranking\_for\_Lowest,

  ROW\_NUMBER() OVER (ORDER BY Avg\_Freight\_Value DESC) AS Ranking\_for\_Highest

FROM

  (SELECT DISTINCT c.customer\_state as State,

    AVG(oi.freight\_value) OVER (PARTITION BY c.customer\_state) AS Avg\_Freight\_Value

  FROM

    `Target.order\_items` AS oi

  INNER JOIN

    `Target.orders` AS o

  USING

    (order\_id)

  INNER JOIN

    `Target.customers` AS c

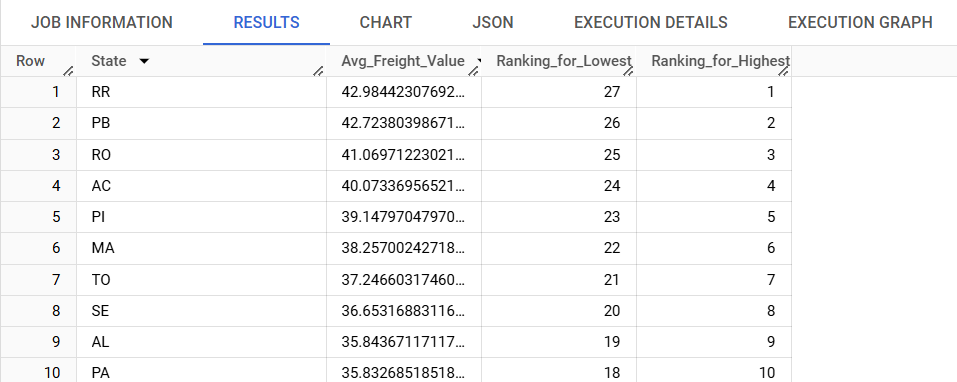
  ON

    o.customer\_id = c.customer\_id) AS T

GROUP BY

  State, Avg\_Freight\_Value;

**🡪 🡪** *Result*:



1. Top 5 states with the lowest average delivery time.

**🡪 🡪** *Query:*

SELECT

  DISTINCT c.customer\_state ,AVG (o.Actual\_Delivery\_Time\_in\_Days) OVER (PARTITION BY c.customer\_state) AS Avg\_Delivery\_Time

FROM

(SELECT

  customer\_id, order\_id,

  TIMESTAMP\_DIFF(order\_delivered\_customer\_date, order\_purchase\_timestamp, day) AS Actual\_Delivery\_Time\_in\_Days ,

  TIMESTAMP\_DIFF(order\_estimated\_delivery\_date, order\_purchase\_timestamp, day) AS Estimated\_Delivery\_Time\_in\_Days,

FROM

  `Target.orders`

WHERE

  order\_status = "delivered") AS o

INNER JOIN

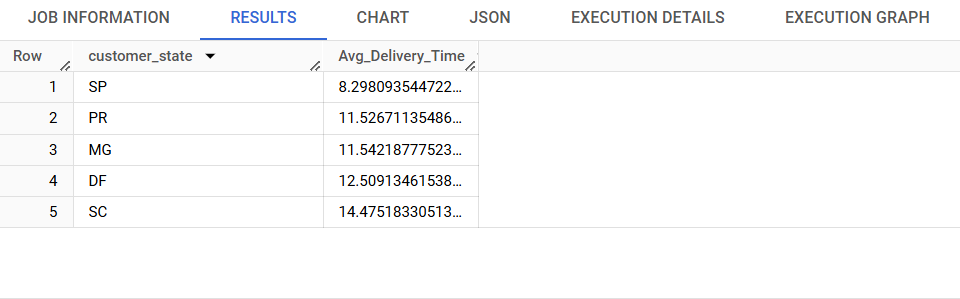
  `Target.customers` AS c

USING

  (customer\_id)

ORDER BY Avg\_Delivery\_Time

LIMIT 5;

**🡪 🡪** *Result*The above data provides top 5 states with lowest average delivery time in days. The data for top 5 states with highest average delivery time can be retrieved by ordering the same data by “Avg\_Delivery\_Time” in descending order.

1. **Analysis based on the payments**
2. Month on month number of orders placed using different payment types.

**🡪 🡪** *Query:*

SELECT

  time\_period,

  payment\_type,

  COUNT(\*) AS Total\_Orders

FROM

  (SELECT

    p.order\_id,

    p.payment\_type,

    EXTRACT(YEAR FROM order\_purchase\_timestamp) AS Year,

    EXTRACT(Month FROM order\_purchase\_timestamp) AS Month,

    FORMAT\_DATE('%b %Y', DATE(order\_purchase\_timestamp)) AS Time\_Period

  FROM

    `Target.payments` AS p

  JOIN

    `Target.orders` AS o

  USING

    (order\_id)) T

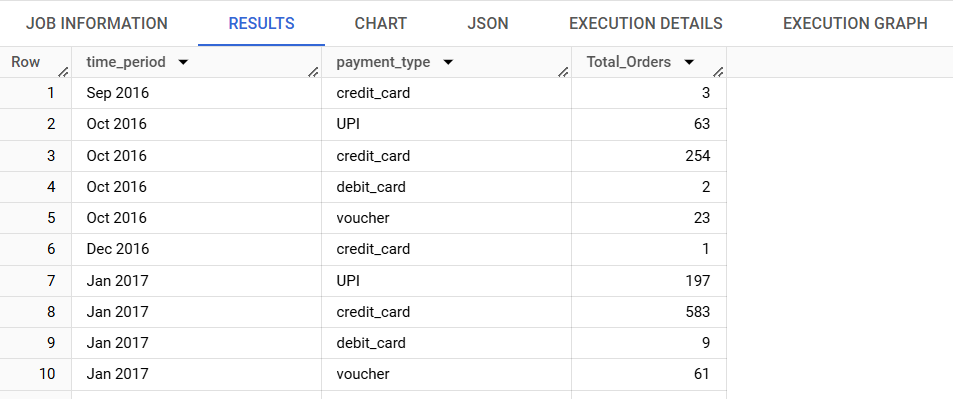
GROUP BY

  time\_period, payment\_type, T.YEAR, T.month

ORDER BY

  T.YEAR, T.month,payment\_type;

**🡪 🡪** *Result*:



1. Count of orders based on the no. of payment instalments.

**🡪 🡪** *Query:*

SELECT

  payment\_installments,

  COUNT(\*) AS Total\_Orders

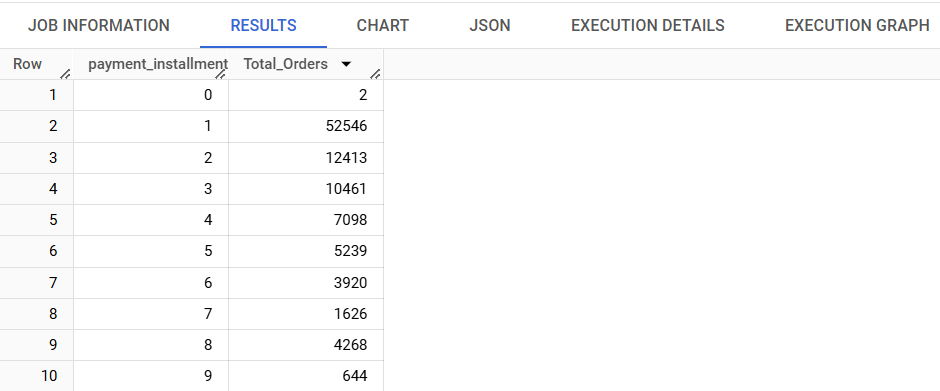
FROM

  `Target.payments`

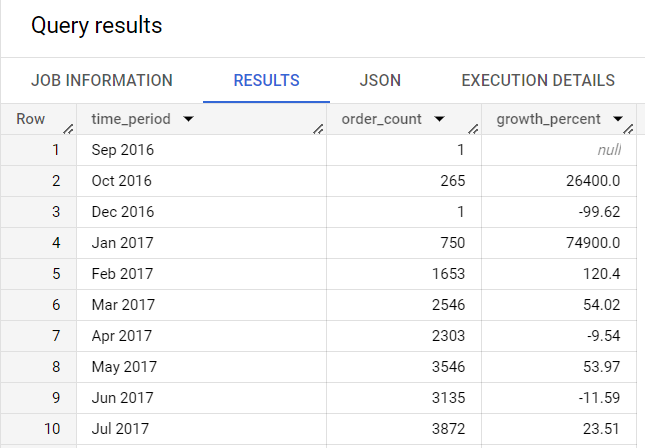
GROUP BY

  payment\_installments;

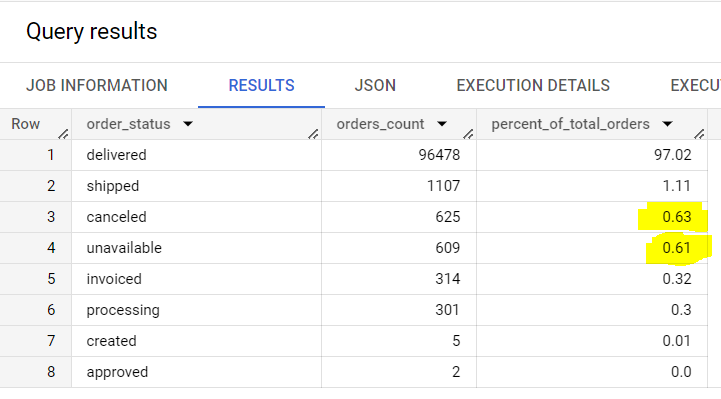
**🡪 🡪** *Result*:



1. **Actionable Insights & Recommendations**
2. The orders trajectory shows a sharp rise in the volume of orders within a short period of time. Business in Brazil is growing quickly, according to the general trend, thus companies need to be prepared with more staff. Company may think about employing contract workers to reduce excessive risk.



1. Total 609 orders were unavailable and 625 orders were cancelled during the given time period, which makes it to be around 1.2 % of total orders. We can reduce this number by studying the reasons behind order cancellation and items unavailability.



1. The query below calculates the ratio of review score for each is state. While, the extracted data is ordered to get the states with higher proportion of unsatisfied customers.

**🡪 🡪** *Query:*

SELECT

  customer\_state AS State,

  ROUND(\_1/Total\_Reviews\*100,1) AS R1,

  ROUND(\_2/Total\_Reviews\*100,1) AS R2,

  ROUND(\_3/Total\_Reviews\*100,1) AS R3,

  ROUND(\_4/Total\_Reviews\*100,1) AS R4,

  ROUND(\_5/Total\_Reviews\*100,1) AS R5

FROM

(SELECT \*,(\_1 + \_2 + \_3 + \_4 +\_5) AS Total\_Reviews

FROM

(SELECT

  \*

FROM (

  SELECT

    c.customer\_state,

    orr.review\_score

  FROM

    `Target.order\_reviews` orr

  JOIN

    `Target.orders` o

  USING

    (order\_id)

  JOIN

    `Target.customers` c

  USING

    (customer\_id))

  PIVOT(COUNT(\*) FOR review\_score IN (1, 2, 3, 4, 5))) AS T)

  ORDER BY R1 DESC,R5;

**🡪 🡪** *Result*:



1. A closer look reveals that the majority of these complaints mention problems with deliveries that were delayed or that the consumer did not get. However, a lot of the negative reviews are also brought on by things that were shipped incorrectly or damaged. Therefore, the business should concentrate on improving its logistics in order to win over customers and increase profitability. The most popular review titles are displayed with the following query.

**🡪 🡪** *Query:*

SELECT

  review\_comment\_title, count(review\_comment\_title) as CNT

FROM

  `Target.order\_reviews`

WHERE

  review\_score IN (1,2)

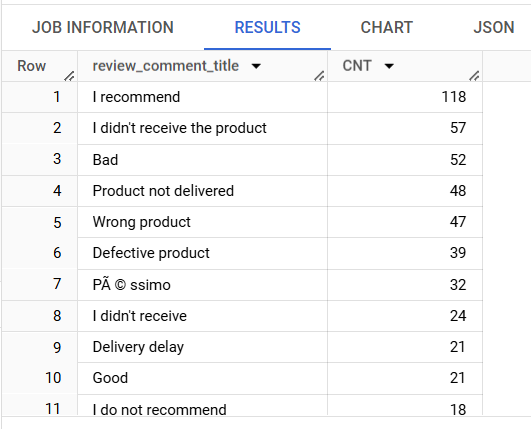
GROUP BY

  review\_comment\_title

ORDER BY

  CNT DESC;

**🡪 🡪** *Result*:



1. Rio de Janeirio, Sao Paulo and Belo Horizonte among others cities in Brazil that contribute the major chunk of orders. The company can upscale their businesses by improving their product offerings and logistics in order to attain higher customer satisfaction to gain more trust in these markets. The results of this analysis can be referred from “In-depth Analysis”, consisting the data for number of customers and number of orders placed per city.
2. Additionally, Brazilian customers show a tendency for shopping online during afternoon and night. The company can focus on scheduling and optimizing their digital marketing campaigns during these hours of the day to increasing customer engagement.