**Business Case: Target SQL**

1. **Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset**
   1. **Data type of columns in a table**

SELECT

 \* EXCEPT(is\_generated,

          generation\_expression,

          is\_stored,

          is\_updatable,

          is\_hidden,

          is\_system\_defined,

          is\_partitioning\_column,

          clustering\_ordinal\_position,

          collation\_name,

          column\_default,

          ordinal\_position,

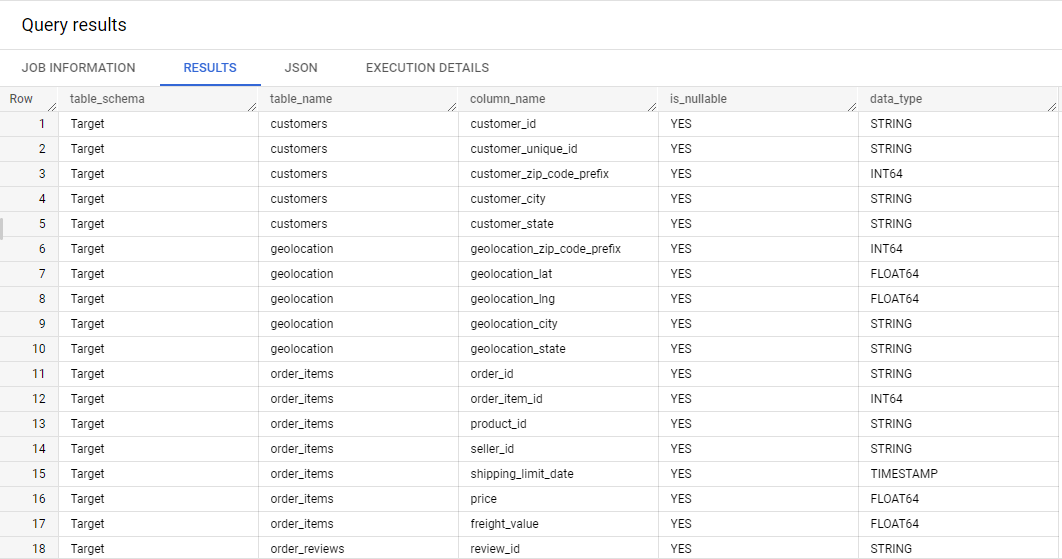
          table\_catalog

        )

 FROM

     `target-sql-362404.Target.INFORMATION\_SCHEMA.COLUMNS`

ORDER BY table\_name;



Count of data types in each table

SELECT a.table\_name,

a.data\_type,

COUNT(a.data\_type) AS count\_of\_datatypes

 FROM  (

         SELECT

               \* EXCEPT(is\_generated,

                        generation\_expression,

                        is\_stored,

                        is\_updatable,

                        is\_hidden,

                        is\_system\_defined,

                        is\_partitioning\_column,

                        clustering\_ordinal\_position,

                        collation\_name,

                        column\_default,

                        ordinal\_position,

                        table\_catalog

                       )

            FROM

                  `target-sql-362404.Target.INFORMATION\_SCHEMA.COLUMNS`

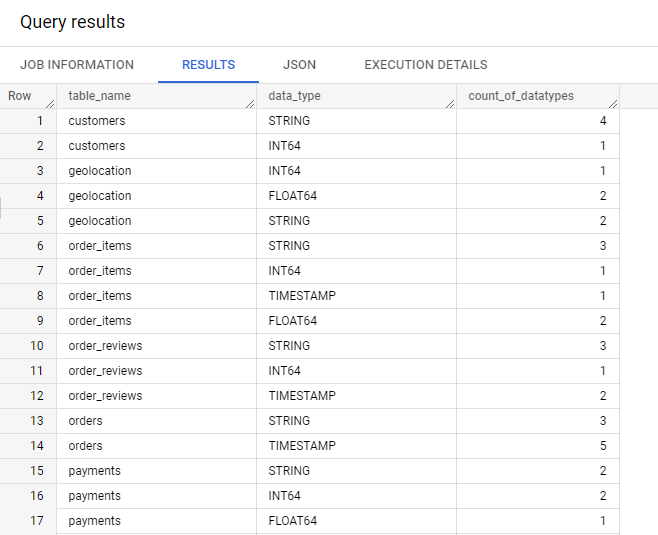
       ) a

  GROUP BY

          a.table\_name, a.data\_type

  ORDER BY

          a.table\_name



* 1. **Time period for which the data is given**

SELECT

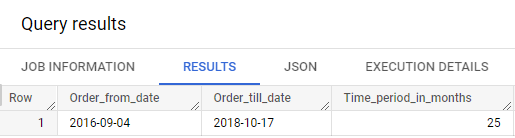
      MIN(DATE(order\_purchase\_timestamp)) AS Order\_from\_date,

      MAX(DATE(order\_delivered\_customer\_date)) AS Order\_till\_date,

      DATE\_DIFF(MAX(DATE(order\_delivered\_customer\_date)),MIN(DATE(order\_purchase\_timestamp)),month) AS Time\_period\_in\_months

  FROM

      target-sql-362404.Target.orders ;



**Insights:**

1. The order data surveyed in the dataset started from **September 2016** till **October 2018**, ie for a time period of 25 months
   1. **Cities and States covered in the dataset**

SELECT COUNT(DISTINCT customer\_id) as Customers,

       COUNT(DISTINCT customer\_unique\_id) AS Unique\_Customer\_ID,

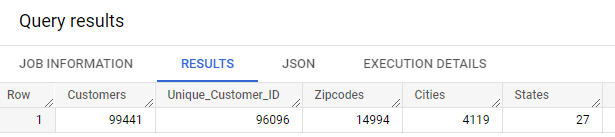
       COUNT(DISTINCT customer\_zip\_code\_prefix) as Zipcodes,

       COUNT(DISTINCT customer\_city) as Cities,

       COUNT(DISTINCT customer\_state) as States

  FROM

       target-sql-362404.Target.customers;



**Insights:**

1. The dataset consists of **99,441 unique customer identifier** who resides in **14,994 Zip location** of Brazil.
2. **There are 4,119 total unique cites within 27 states from which orders have been placed**

**# Customers per state Analysis:**

SELECT

      customer\_state,

      COUNT(customer\_id) as number\_of\_customers

FROM

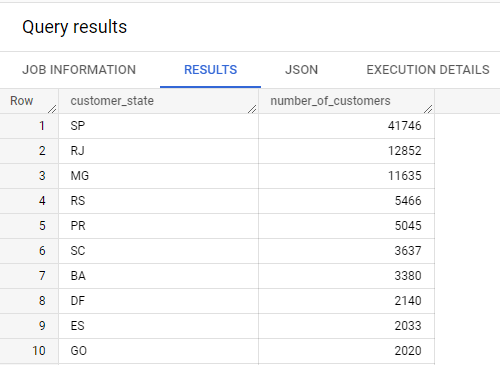
    `target-sql-362404.Target.customers`

GROUP BY

      customer\_state

ORDER BY

      number\_of\_customers DESC;

****

1. **In-depth Exploration:**
   1. **Is there a growing trend on e-commerce in Brazil? How can we describe a complete scenario? Can we see some seasonality with peaks at specific months?**
2. **Revenue per year**

SELECT

      EXTRACT(YEAR FROM order\_purchase\_timestamp) as year\_of\_purchase,

      ROUND(SUM(price\*freight\_value),2) AS revenue,

      COUNT(o.order\_id) as number\_of\_orders

FROM

    `target-sql-362404.Target.orders` as o

JOIN

    `target-sql-362404.Target.order\_items` as oi

ON

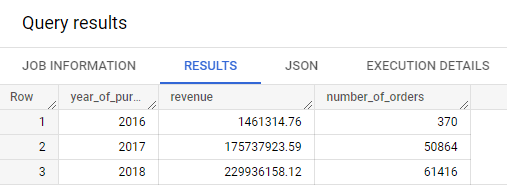
    o.order\_id = oi.order\_id

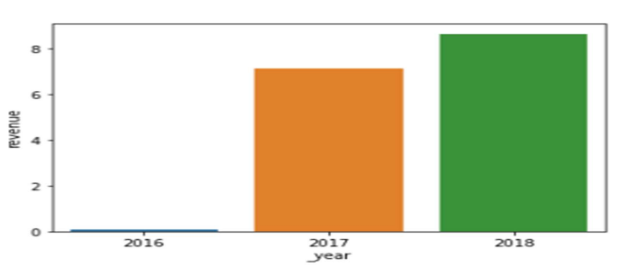
GROUP BY

    year\_of\_purchase

ORDER BY

    year\_of\_purchase ;





**Insights:**

1) In comparison to **2017, sales revenue** has **increased** in **2018 by 21%**

**b) Average order per Month**

SELECT

      x.month,

      AVG(x.num\_of\_orders) AS avgerage\_orders\_per\_month

FROM

(

  SELECT

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

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

    COUNT(order\_id) as num\_of\_orders

  FROM

      `target-sql-362404.Target.orders`

  GROUP BY

      year,

      month

  ORDER BY

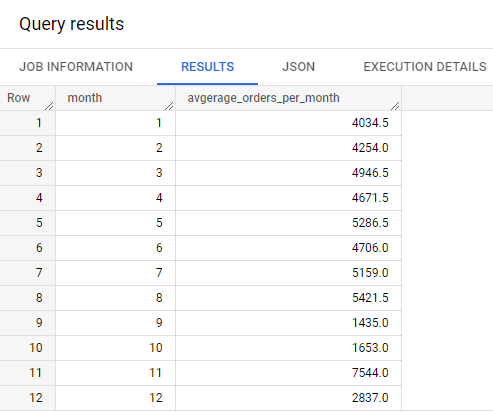
      year,

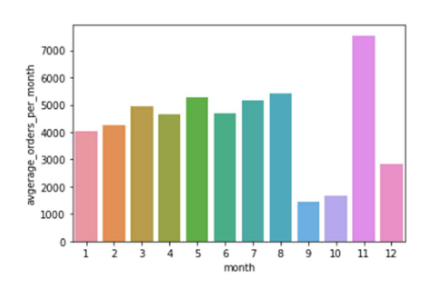
      month

) as x

GROUP BY x.month

ORDER BY x.month ;





**Insights:**

1) In the month of **November** the average number of order placed are **highest**

2) **September and October** month average orders are **comparatively low**.

3) In **May, July and August** order placed are **higher** **compared to other months**

SELECT DISTINCT

               EXTRACT(MONTH FROM o.order\_delivered\_carrier\_date) as purchase\_month,

               EXTRACT(YEAR FROM o.order\_delivered\_carrier\_date) as purchase\_year,

               SUM(payment\_value) OVER(PARTITION BY

                                         EXTRACT(YEAR FROM o.order\_delivered\_carrier\_date),

                                         EXTRACT(MONTH FROM o.order\_delivered\_carrier\_date)

                                       ORDER BY

                                         EXTRACT(YEAR FROM o.order\_delivered\_carrier\_date),

                                         EXTRACT(MONTH FROM o.order\_delivered\_carrier\_date)                                  RANGE BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING

                                      ) AS total\_payment

FROM

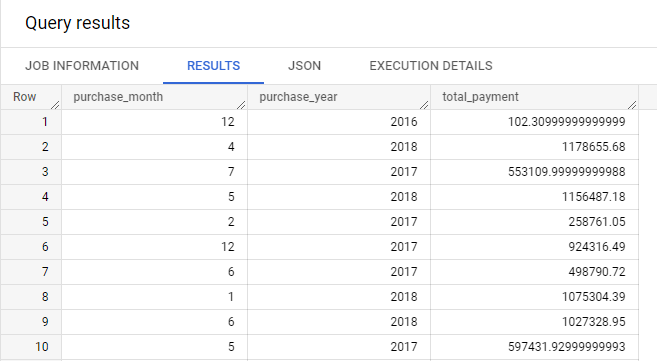
     `target-sql-362404.Target.orders` o

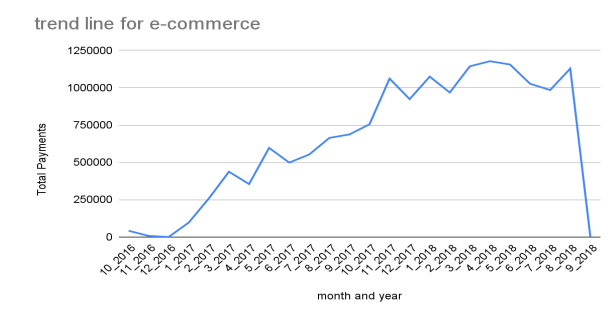
LEFT JOIN

     `target-sql-362404.Target.payments` p

ON

     o.order\_id = p.order\_id;





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

**a) Number of Orders per Hour**

SELECT

      X.HOUR,

      COUNT(X.order\_id) AS number\_of\_orders

FROM

 (

 SELECT

       order\_id,

       EXTRACT (HOUR FROM order\_purchase\_timestamp) AS HOUR

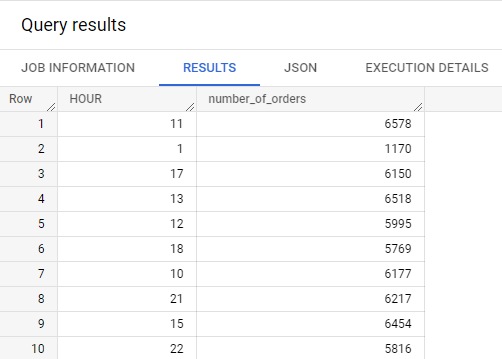
 FROM

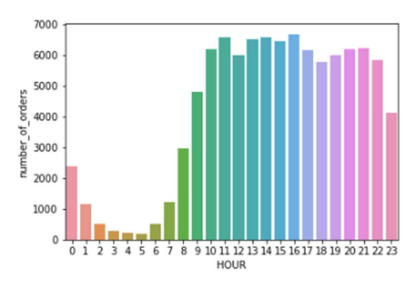
     `target-sql-362404.Target.orders`

 ) AS X

GROUP BY

      X.HOUR;

****

****

**b) Number of orders per hour during daytime**

SELECT

      X.HOUR,

      COUNT(X.order\_id) AS number\_of\_orders,

      CASE

          WHEN X.HOUR BETWEEN 5 AND 7 THEN 'Dawn'

          WHEN X.HOUR BETWEEN 8 AND 11 THEN 'Morning'

          WHEN X.hour BETWEEN 12 AND 17 THEN 'Afternoon'

          WHEN X.hour BETWEEN 18 and 21 THEN 'Evening'

          ELSE 'night'

      END AS time

FROM

 (

 SELECT

       order\_id,

       EXTRACT (HOUR FROM order\_purchase\_timestamp) AS HOUR

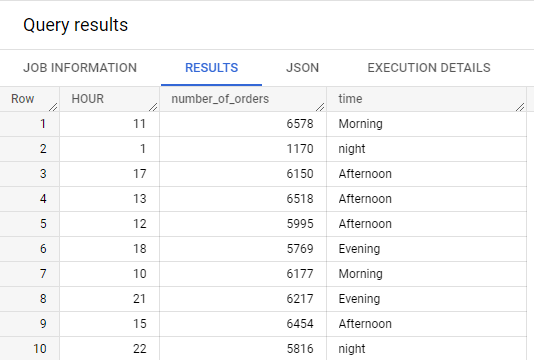
 FROM

     `target-sql-362404.Target.orders`

 ) AS X

GROUP BY

      X.HOUR;

****

SELECT DISTINCT

               daytime\_info.daytime AS Daytime,

               SUM(payment\_value) OVER (PARTITION BY daytime

       ORDER BY daytime

) As Total\_Payment

FROM

(SELECT

    p.payment\_value,

    CASE

           WHEN EXTRACT (HOUR FROM o.order\_delivered\_carrier\_date) BETWEEN 0 AND 6

       THEN "Dawn"

           WHEN EXTRACT (HOUR FROM o.order\_delivered\_carrier\_date) BETWEEN 7 AND 11

      THEN "Morning"

          WHEN EXTRACT (HOUR FROM o.order\_delivered\_carrier\_date) BETWEEN 12 AND 17

      THEN "Afternoon"

          WHEN EXTRACT (HOUR FROM o.order\_delivered\_carrier\_date) BETWEEN 18 AND 23

      THEN "Night"

    END AS Daytime

FROM

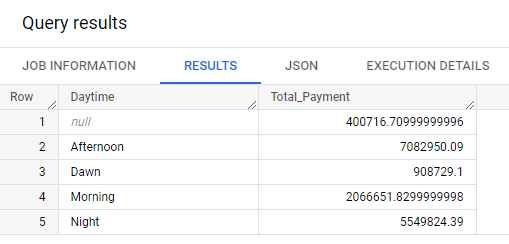
    `target-sql-362404.Target.orders` o

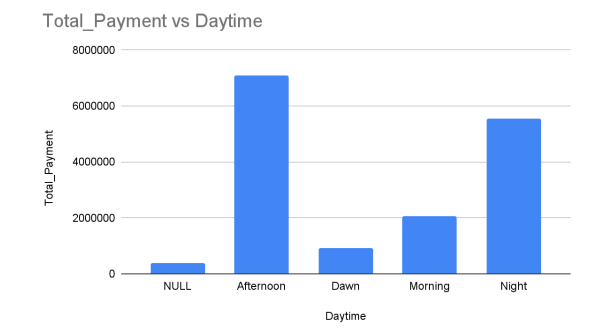
LEFT JOIN

    `target-sql-362404.Target.payments` p

ON

    o.order\_id = p.order\_id) As daytime\_info;

****

****

**Insights:**

1) Customers are generally purchasing during **Moring at 8:00 AM till late Evening upto 11:00 PM**

2) **Afternoon orders** have a **huge spike** and also **Evening** , compared to morning , and night hour

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

SELECT DISTINCT

              EXTRACT (YEAR FROM o.order\_delivered\_carrier\_date) AS purchase\_year,

             EXTRACT (MONTH FROM o.order\_delivered\_carrier\_date) AS purchase\_month,

              COUNT(o.order\_id) OVER( PARTITION BY

                                       EXTRACT (YEAR FROM o.order\_delivered\_carrier\_date),                                        EXTRACT (MONTH FROM o.order\_delivered\_carrier\_date)

                                          ORDER BY                                             EXTRACT (YEAR FROM o.order\_delivered\_carrier\_date) ,

                              EXTRACT (MONTH FROM o.order\_delivered\_carrier\_date)

                                   ) As No\_of\_orders

FROM

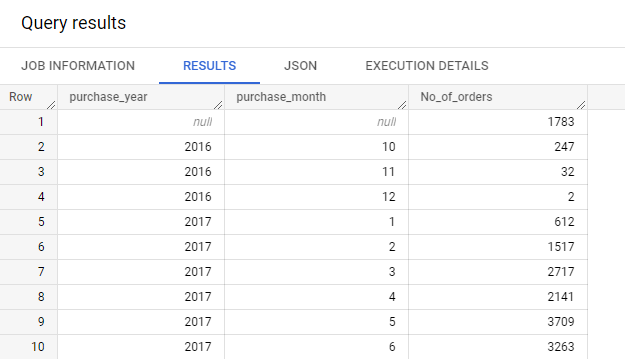
    `target-sql-362404.Target.orders` o

LEFT jOIN

    `target-sql-362404.Target.customers` c

ON

  o.customer\_id= c.customer\_id;



* 1. **How are customers distributed in Brazil**

SELECT DISTINCT customer\_state,

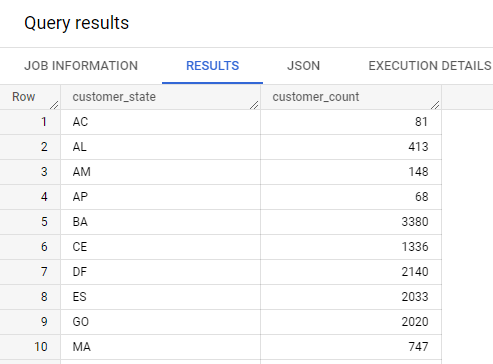
       COUNT(customer\_id) OVER(PARTITION BY customer\_state ORDER BY customer\_state                           RANGE BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING

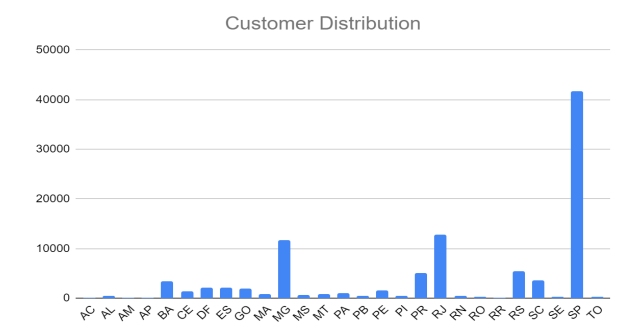
                              ) as customer\_count

FROM

    `target-sql-362404.Target.customers`

ORDER BY customer\_state;

****

****

**Insights:**

**1) 99441 customers** are there in given data:

a) **68% customers** are from **southeast Brazil**

b**) 14 Total %** are from **south Brazil.**

c) Rest are from **other regions of Brazil**

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

SELECT

      \*,

      (Total\_Cost\_2018-Total\_Cost\_2017)/Total\_Cost\_2017 \* 100 As Precentage\_Increase

FROM

  (SELECT

        Purchase\_Month,

        Max(Total\_Cost\_2017) As Total\_Cost\_2017,

        Max(Total\_Cost\_2018) As Total\_Cost\_2018

   FROM

     (SELECT

           Purchase\_Month,

           CASE

               WHEN purchase\_year=2017

                    THEN TotalCost

               END AS Total\_Cost\_2017,

           CASE

               WHEN purchase\_year=2018

                    THEN TotalCost

               END AS Total\_Cost\_2018

      FROM

        (SELECT

              DISTINCT

                  FORMAT\_DATE('%B',order\_delivered\_carrier\_date) AS Purchase\_Month,

              EXTRACT (YEAR FROM order\_delivered\_carrier\_date) AS Purchase\_Year,

              SUM(oi.price) OVER ( PARTITION BY

                                  EXTRACT(MONTH FROM order\_delivered\_carrier\_date),

                                    EXTRACT(YEAR FROM order\_delivered\_carrier\_date)

                                   ORDER BY

                                  EXTRACT(MONTH FROM order\_delivered\_carrier\_date),

                                     EXTRACT(YEAR FROM order\_delivered\_carrier\_date)

                          RANGE BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING

                                  )AS TotalCost

          FROM target-sql-362404.Target.orders o

        LEFT JOIN

               target-sql-362404.Target.order\_items oi

             ON o.order\_id=oi.order\_id

        LEFT JOIN target-sql-362404.Target.payments p

             ON p.order\_id=o.order\_id

    WHERE

           EXTRACT(MONTH FROM order\_delivered\_carrier\_date) BETWEEN 1 AND 8

       AND EXTRACT(YEAR FROM order\_delivered\_carrier\_date) IN (2017,2018)

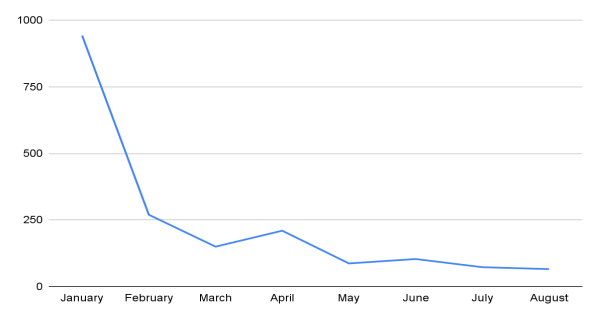
) as month\_wise\_data) as year\_wise\_data

GROUP BY

    Purchase\_Month

) AS Year\_Month\_Data;

****

****

* 1. **Mean & Sum of price and freight value by customer state**

SELECT

      customer\_state,

      AVG(oi.price) AS mean\_price,

      AVG(oi.freight\_value) AS mean\_freight

  FROM

    target-sql-362404.Target.order\_items oi

  JOIN

    target-sql-362404.Target.orders o

  ON o.order\_id=oi.order\_id

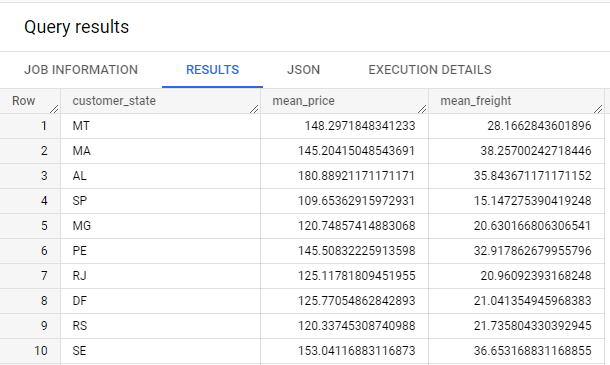
  JOIN

    target-sql-362404.Target.customers c

  ON c.customer\_id=o.customer\_id

GROUP BY

     c.customer\_state;

****

****

1. **Analysis on sales, freight and delivery time**
   1. **Calculate days between purchasing, delivering and estimated delivery**
   2. **Create columns:**
      * time\_to\_delivery = order\_purchase\_timestamp-order\_delivered\_customer\_date
      * diff\_estimated\_delivery = order\_estimated\_delivery\_date-order\_delivered\_customer\_date

SELECT

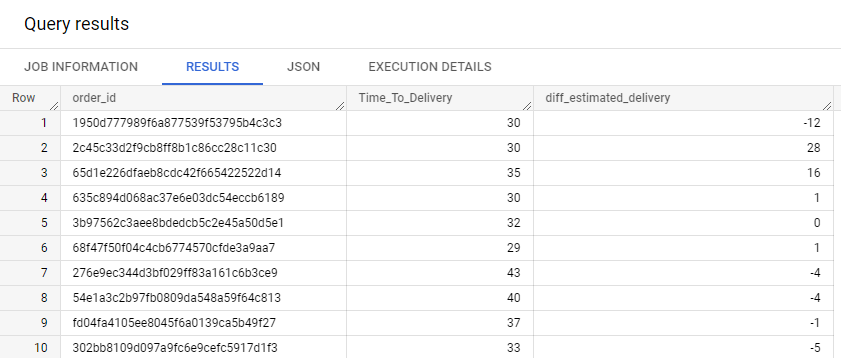
      order\_id,

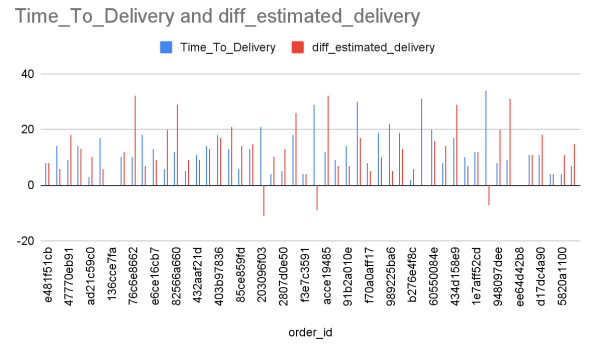
      DATE\_DIFF(order\_delivered\_customer\_date,order\_purchase\_timestamp,day) AS Time\_To\_Delivery,

      DATE\_DIFF(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,day) AS diff\_estimated\_delivery

FROM

      `target-sql-362404.Target.orders`;





* 1. **Group data by state, take mean of freight\_value, time\_to\_delivery, diff\_estimated\_delivery**

**# Analysis on Delivery time**

* What is the expected time for delivery ?
* By how many days delivery is exceeded from expected delivery time ?
* Was the delivery made on time ?

SELECT

 order\_id,

 date\_diff(order\_delivered\_customer\_date,order\_estimated\_delivery\_date,day) as days\_exceeded\_from\_expected\_delivery,

 date\_diff(order\_delivered\_customer\_date,order\_purchase\_timestamp,day) as days\_taken\_to\_delivery,

 date\_diff(order\_estimated\_delivery\_date,order\_purchase\_timestamp,day) as estimated\_time\_to\_delivery

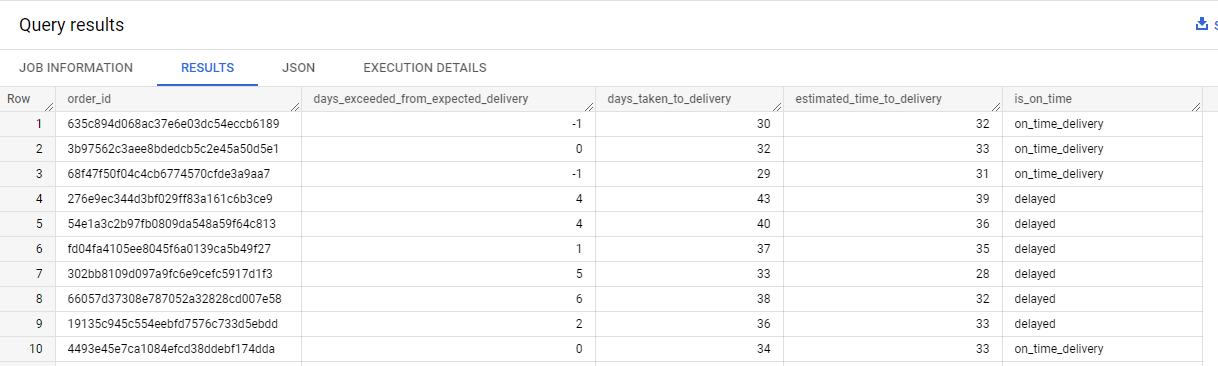
FROM

  target-sql-362404.Target.orders

WHERE

 order\_status = 'delivered'

)



SELECT

     DISTINCT

     x.delivery\_on\_time,

     COUNT(\*) OVER (PARTITION BY x.delivery\_on\_time) AS count\_of\_records

FROM

   (

    SELECT

          \*,

          CASE

              WHEN order\_delivered\_customer\_date IS NULL THEN 'not\_yet\_delivered'

              WHEN (order\_delivered\_customer\_date < order\_estimated\_delivery\_date) THEN 'on\_time\_delivery'

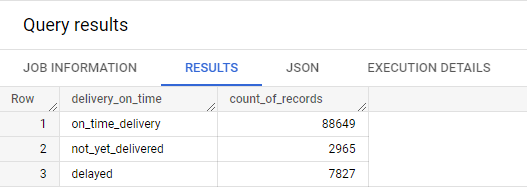
                  ELSE 'delayed'

          END AS delivery\_on\_time

      FROM

          `target-sql-362404.Target.orders`

   ) AS x ;



SELECT

      order\_id,

      order\_status,

      date\_diff(order\_approved\_at,order\_purchase\_timestamp,day) as approval\_time,

      date\_diff(order\_delivered\_carrier\_date,order\_approved\_at,day) as time\_taken\_to\_start\_delivery\_by\_carrier,

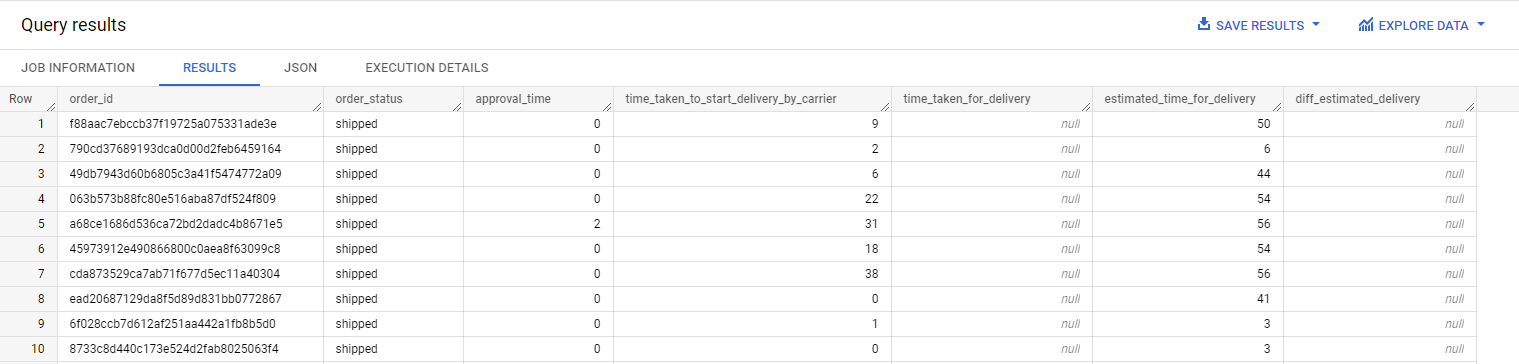
      date\_diff(order\_delivered\_customer\_date,order\_purchase\_timestamp,day) as time\_taken\_for\_delivery,

      date\_diff(order\_estimated\_delivery\_date,order\_purchase\_timestamp,day) as estimated\_time\_for\_delivery,

      date\_diff(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,day) as diff\_estimated\_delivery

FROM

          `target-sql-362404.Target.orders`;



SELECT

 c.customer\_state,

 avg(date\_diff(order\_approved\_at,order\_purchase\_timestamp,day)) as mean\_of\_approval\_time,

 avg(date\_diff(order\_delivered\_carrier\_date,order\_approved\_at,day)) as mean\_of\_time\_taken\_to\_start\_delivery\_by\_carrier,

 avg(date\_diff(order\_delivered\_customer\_date,order\_purchase\_timestamp,day)) as mean\_of\_time\_taken\_for\_delivery,

 avg(date\_diff(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,day)) as mean\_of\_diff\_estimated\_delivery,

 avg(ois.freight\_value) avg\_fright\_value

FROM

`target-sql-362404.Target.orders` as o

JOIN

`target-sql-362404.Target.customers` as c

on o.customer\_id = c.customer\_id

JOIN

`target-sql-362404.Target.order\_items` as ois

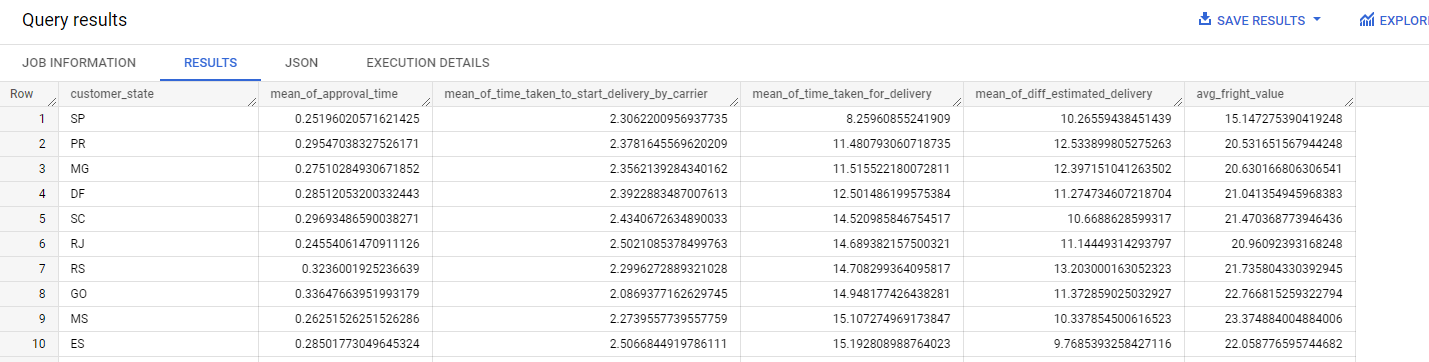
on ois.order\_id = o.order\_id

group by

c.customer\_state

order by

mean\_of\_time\_taken\_for\_delivery ;



SELECT

 c.customer\_state,

 avg(ois.freight\_value) avg\_fright\_value,

 avg(date\_diff(order\_delivered\_customer\_date,order\_purchase\_timestamp,day)) as time\_taken\_for\_delivery,

 avg(date\_diff(order\_estimated\_delivery\_date,order\_purchase\_timestamp,day)) as estimated\_time\_for\_delivery,

 avg(date\_diff(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,day)) as diff\_estimated\_delivery,

 avg(date\_diff(order\_delivered\_carrier\_date,order\_approved\_at,day)) as time\_taken\_to\_start\_delivery\_by\_carrier,

 avg(date\_diff(order\_approved\_at,order\_purchase\_timestamp,day)) as approval\_time

FROM

`target-sql-362404.Target.orders` as o

JOIN

`target-sql-362404.Target.customers` as c

on o.customer\_id = c.customer\_id

JOIN

`target-sql-362404.Target.order\_items` as ois

on ois.order\_id = o.order\_id

WHERE

 o.order\_delivered\_customer\_date is not null and

 o.order\_delivered\_carrier\_date is not null and

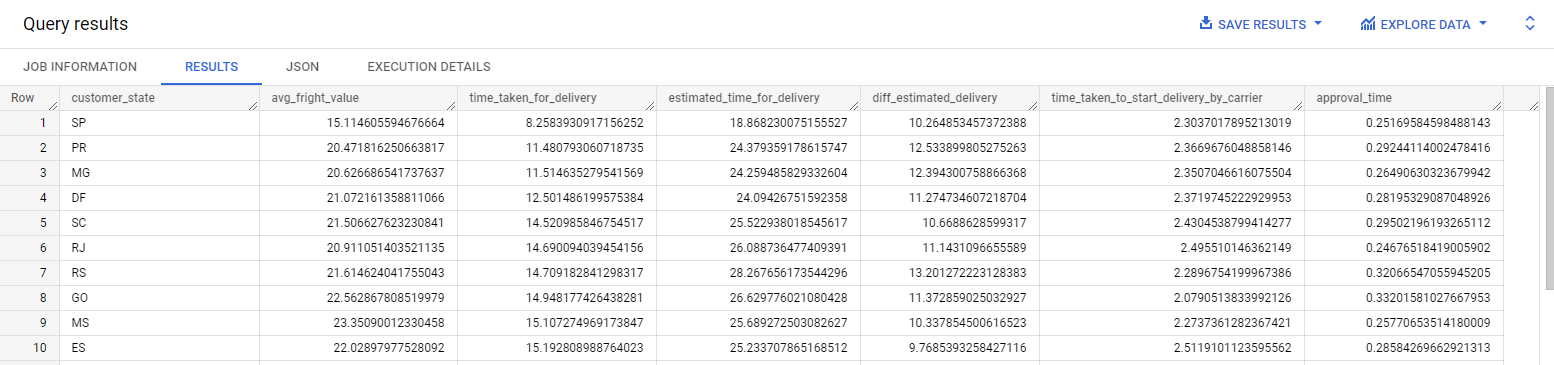
 o.order\_approved\_at is not null

GROUP BY

 c.customer\_state

ORDER BY

 time\_taken\_for\_delivery;



SELECT

      c.customer\_state,

      AVG(oi.freight\_value) AS mean\_freight\_value,

      AVG(DATE\_DIFF(order\_delivered\_customer\_date,order\_purchase\_timestamp,day)) as Mean\_Time\_To\_Delivery,

      AVG(DATE\_DIFF(order\_estimated\_delivery\_date,order\_delivered\_customer\_date,day)) as AVG\_diff\_estimated\_delivery

FROM

     `target-sql-362404.Target.order\_items` oi

JOIN

     `target-sql-362404.Target.orders` o

    ON o.order\_id=oi.order\_id

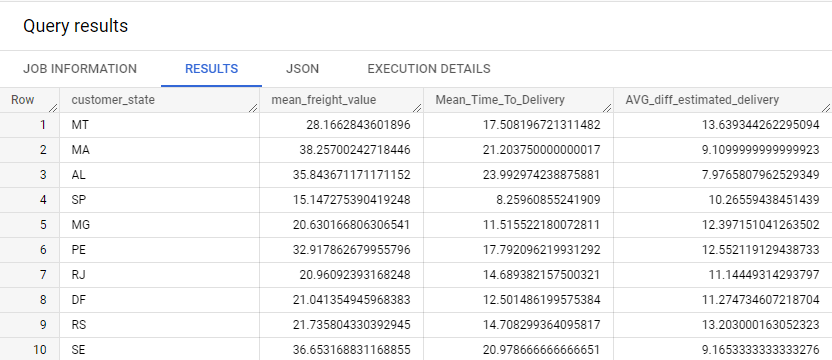
JOIN

     `target-sql-362404.Target.customers` c

    ON c.customer\_id=o.customer\_id

GROUP BY

    c.customer\_state;



**Insights:**

1) **After purchase being made**, the **average time for approving** the order **by seller** is **0.26** days and median time is 0 , **means with in a day**.

2) **Average time** taken for a **carrier to start the delivery is 2 and a half day**.

3) **Average time** taken to **complete delivery is 12 days** and **median of delivery time is 10 days.**

4) Estimated **time delivery average** is **23 days**.

5) There is a **positive correlation** between **freight value and delivery time**.

6) **Long distance deliveries** are having **higher freight values** and also takes **more time for delivery**

7) **States São Paulo ,Paraná,Minas Gerais, Distrito Federal ,Santa Catarina and Rio de Janeiro** are some of the states having relatively **faster delivery time.**

8) **Alagoas, Amazonas, Amapá ,Pará and Roraima** are some states have relatively **very slow delivery time.**

* 1. **Sort the data to get the following:**

1. Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5

SELECT

      c.customer\_state,

      AVG(oi.freight\_value) AS mean\_freight\_value

FROM

     `target-sql-362404.Target.order\_items` oi

JOIN

     `target-sql-362404.Target.orders` o

    ON o.order\_id=oi.order\_id

JOIN `target-sql-362404.Target.customers` c

    ON c.customer\_id=o.customer\_id

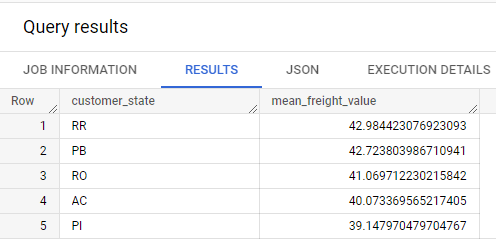
GROUP BY

    c.customer\_state

ORDER BY

    mean\_freight\_value DESC

LIMIT 5;



* + - Top 5 states with highest/lowest average time to delivery

SELECT

      c.customer\_state,

      AVG(DATE\_DIFF(order\_delivered\_customer\_date,order\_delivered\_carrier\_date,day) as Avg\_Time\_To\_Delivery

FROM

     `target-sql-362404.Target.order\_items` oi

JOIN

     `target-sql-362404.Target.orders` o

    ON o.order\_id=oi.order\_id

JOIN `target-sql-362404.Target.customers` c

    ON c.customer\_id=o.customer\_id

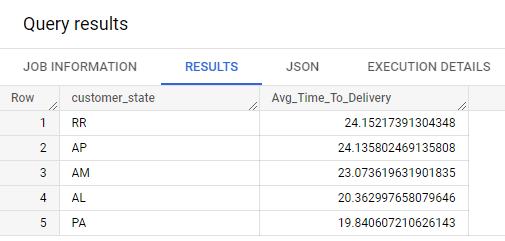
GROUP BY

    c.customer\_state

ORDER BY

    Avg\_Time\_To\_Delivery DESC

LIMIT 5;



* + - Top 5 states where delivery is really fast/ not so fast compared to estimated date

SELECT

     c.customer\_state,

     AVG(DATE\_DIFF(order\_estimated\_delivery\_date,order\_delivered\_carrier\_date,day) as Mean\_Expected\_Time\_To\_Delivery

FROM

     `target-sql-362404.Target.order\_items` oi

JOIN

     `target-sql-362404.Target.orders` o

    ON o.order\_id=oi.order\_id

JOIN `target-sql-362404.Target.customers` c

    ON c.customer\_id=o.customer\_id

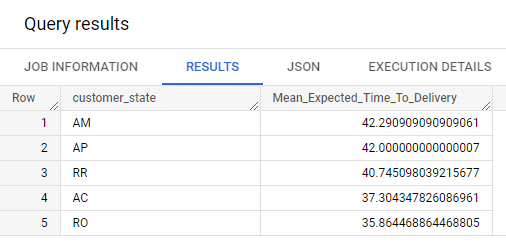
GROUP BY

    c.customer\_state

ORDER BY

    Mean\_Expected\_Time\_To\_Delivery DESC

LIMIT 5;



1. **Payment type analysis:**

**# Analysing Payment details:**

SELECT

 \* EXCEPT(is\_generated, generation\_expression, is\_stored, is\_updatable,is\_hidden

        ,is\_system\_defined,is\_partitioning\_column,

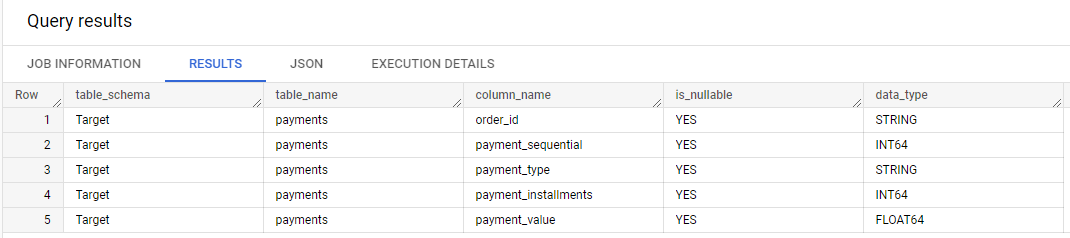
         clustering\_ordinal\_position,collation\_name,column\_default,ordinal\_position,table\_catalog)

 FROM

 `target-sql-362404.Target.INFORMATION\_SCHEMA.COLUMNS`

 WHERE

 table\_name = 'payments';

****

SELECT

      payment\_type,

      COUNT(DISTINCT(order\_id)) AS Number\_of\_sales\_per\_payment\_type,

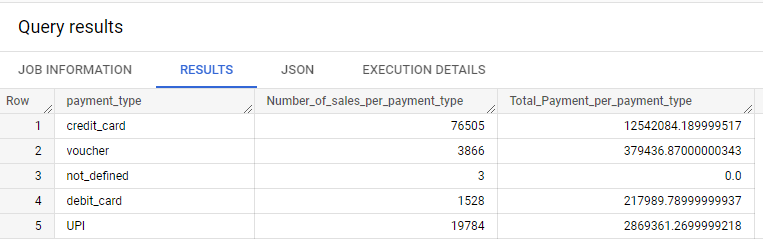
      SUM(payment\_value) AS Total\_Payment\_per\_payment\_type

FROM

     `target-sql-362404.Target.payments`

GROUP BY

        payment\_type;

****

**Insights:**

1) **78% payments** are done using **credit card** and **17.92%** are done with **UPI**.

* 1. **Month over Month count of orders for different payment types**

SELECT

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

      FORMAT\_TIMESTAMP("%b %Y", order\_purchase\_timestamp) AS Month\_year\_purchase\_date,

      p.Payment\_Type,

      COUNT(o.order\_id) AS Number\_of\_Orders

FROM

     `target-sql-362404.Target.payments` as p

JOIN

     `target-sql-362404.Target.orders` as o

    ON o.order\_id = p.order\_id

GROUP BY

    Year,

    Month\_year\_purchase\_date,

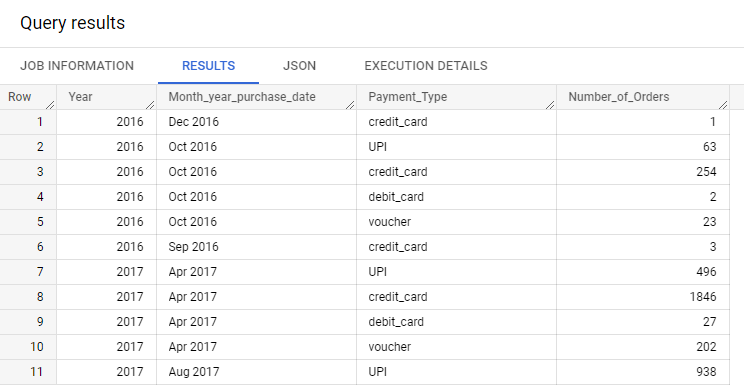
    p.payment\_type

ORDER BY

    Year,

    Month\_year\_purchase\_date,

    p.payment\_type;

****

* 1. **Distribution of payment installments and count of orders**

SELECT

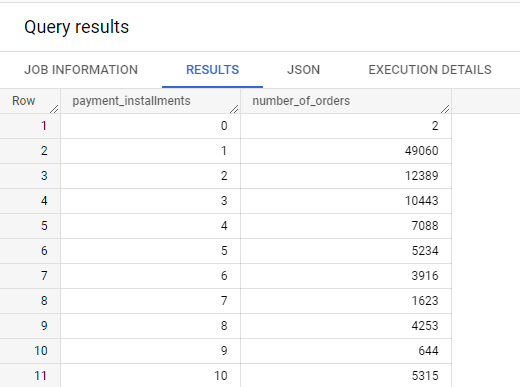
      payment\_installments,

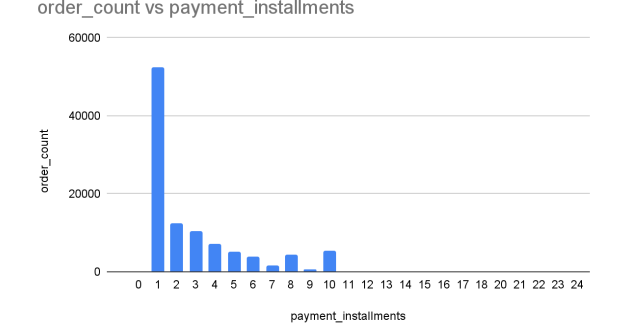
      COUNT(DISTINCT(order\_id)) AS number\_of\_orders

FROM

    `target-sql-362404.Target.payments`

GROUP BY payment\_installments ;





**Insights:**

**1) Majority** of the orders are purchased with **1 payment instalment.**

**2)** Also **more than 5 instalments purchases are relatively very low.**

**Insights & Recommendation:**

**All Insights :**

1) Customers'**information’s:**

        We have 99,441 customers of data available, out of which 96096 Unique Customers IDs with 14994 different locations of customers

        Customers are from different 4119 cities and 27 states in Brazil

        From a total of 99441 orders, 1107 are shipped, 625 were cancelled, and 96478 are delivered

        68% of customers are from southeast Brazil, 14% are from south Brazil and the rest are from other regions of Brazil

2) **Analysis of sales and revenue as per time:**

        Time period for which the data is given is 25 months

        As compared to 2017 the revenue has increased in 2018 by 21%

       The Average number of orders is higher during the month of November. September and October have comparatively low orders on average whereas May, July, and August have a higher number of average orders compared to other months.

        Tuesday, Monday and Wednesdays have a relatively higher number of orders

3) **Increasing trend:**

        There is an increasing trend in orders, trend sustains during 2018. There is a slight fall we can observe during October 2017 followed by a great hike in November month and again a fall at end of December 2017 and January 2018.

        We can observe the trend of increasing orders with time and also for revenue.

        We can observe there's an 81.5% growth increase in terms of orders and a 70.7% growth increment in terms of revenue in January from 2017 to 2018.

        Growth rate for July and August from 2017 to 2018 is relatively very low whereas2017-February, march, and November were the highest growing sale month in comparison to the previous month.

4) **Customer\_purchasing Behavior:**

        Customers are purchasing orders from morning 8 am till late evening 11 pm.

        Afternoon and evening orders are very high as compared to the morning, and night time.

5) **Delivery time:**

        After the purchase is made, the average time for approving the order by the seller is 0.26 days and the median time is 0, which means within a day.

        Average time taken for a carrier to start the delivery is 2 and a half days.

        Average time taken to complete delivery is 12 days and the median delivery time is 10 days.

        Estimated time delivery average is 23 days.

        There is a positive correlation between freight value and delivery time.

        Long-distance deliveries are having higher freight values and also take more time for delivery

        States São Paulo, Paraná, Minas Gerais, Distrito Federal, Santa Catarina, and Rio de Janeiro are some of the states having relatively faster delivery times.

        Alagoas, Amazonas, Amapá, Pará, and Roraima are some states that have relatively very slow delivery times.

6) **Region and State-wise Analysis :**

        São Paulo, Rio de Janeiro, Minas Gerais, Rio Grande do Sul, and Paraná are the top 5 highest orders states and also generate the highest revenue.

        More than 80% of orders are coming from south, southeast, and northeast Brazil. 90% of the revenue is coming from south, southeast, and northeast Brazil

7) **Payment type-related info :**

        78% of payments are done using a credit card and 17.92% are done with UPI.

        Majority of the orders are purchased at 1 payment installment.

        More than 5 installments purchases are relatively very low.

**Recommendations:**

1.      From the distribution and statistical analysis we can observe the average time to complete the delivery is 12 days. which should be reduced to at least half, as due to high competition in the e-commerce market, it is vital to do so

2.      In order to reduce the delivery time, if we look at the average time for the carrier to start the delivery itself takes at least 2 and a half days and the order approval time is 0.26 days. These two should be optimized as low as possible, which can result in delivery faster.

3.      If we look at the Top states where delivery is really slow compared to the estimated date, they are all from the north Brazil region. Delivering faster in the northern states may create and increase new customers and revenue from the north.

4.      As per Analysis products belonging to the “bed table bath” category are being sold max among all available categories, we could produce more items related to this category.

5.      Increasing the network in north Brazil, having small towns can help increase the customer base. As north Brazil has the world’s largest river and most extensive rain forest, must be a good travel destination, introducing necessary survival/ camping/adventure products can help increase revenue and order from the northern region

6.      As per the analysis Credit card payments are more. So target can give offers to customers paying with a credit card and can also give them an Interest-free EMI scheme.

7.      It was observed that an increasing trend in revenue and orders over time, yet during October and January sales are decreasing probably after Festival Sales. Introducing possible discounts on the not-so-running products can help sell more products during those low-going months.