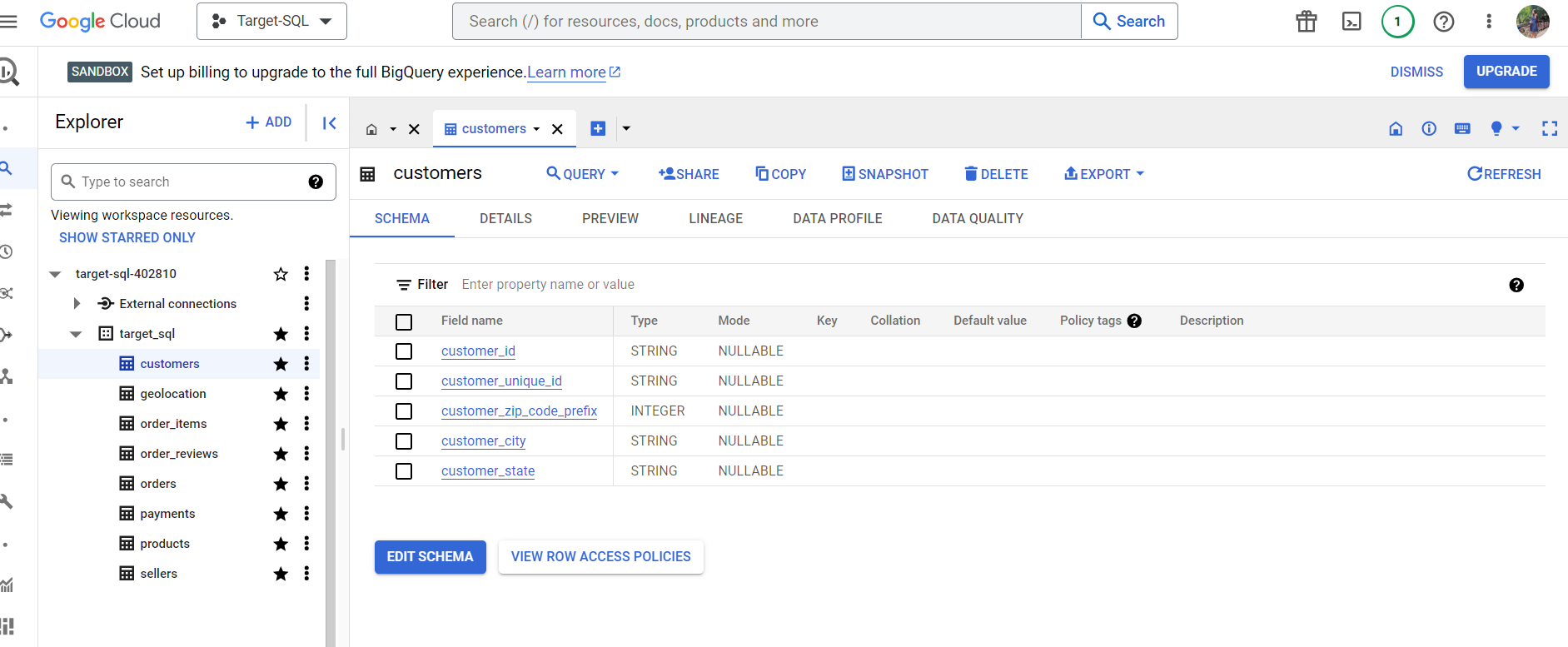
1. Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset:
   1. Data type of all columns in the "customers" table.

**Solution** – you can use the open option available in the query to see the data type of all

Columns in customer table. You have click on the table’s icon and you find the open option to see the data type of all Columns in customer table.

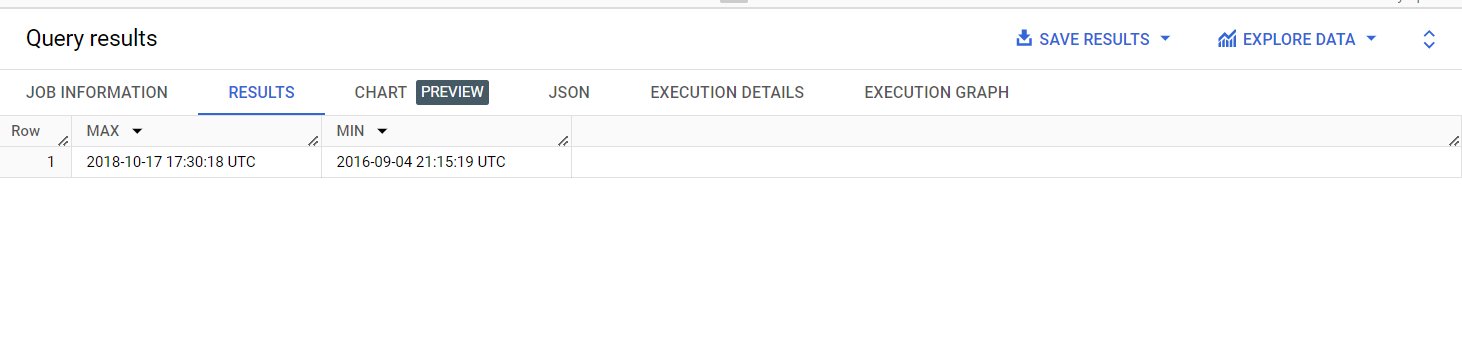


* 1. Get the time range between which the orders were placed.

**Solution** - select max(order\_purchase\_timestamp

) as MAX,min(order\_purchase\_timestamp

) as MIN from `target\_sql.orders`



Observations – I have observed that order start from 4th of September 2016 and last order was placed on 17th of October 2018 all the order placed between this time frame.

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

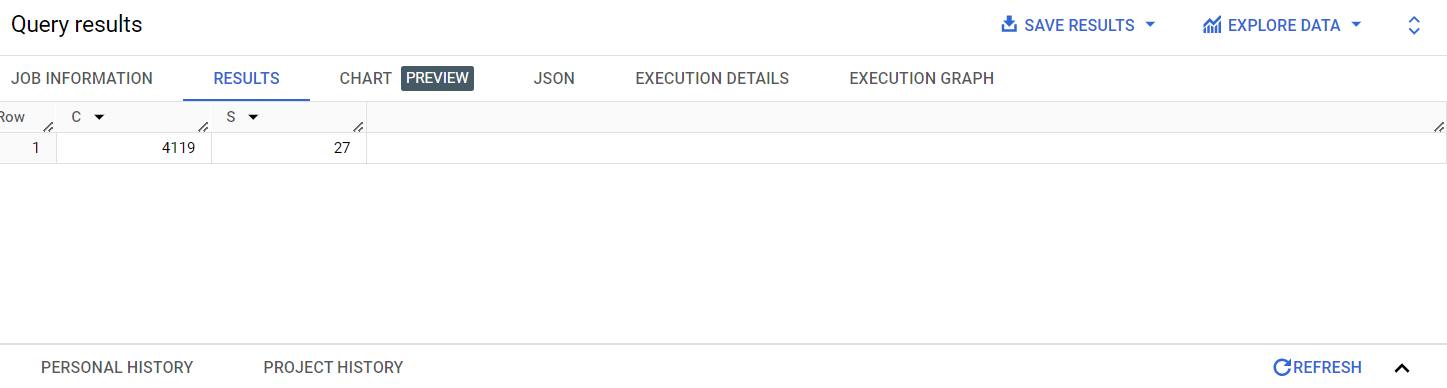
Solution -: select count(distinct customer\_city) as C,

count (distinct customer\_state) as S

 from `target\_sql.customers` as T

inner join `target\_sql.orders` as O

on T.customer\_id = O.customer\_id;



Observations – I have observed that during the given time period from 4119 unique cities and 27 unique states orders placed.

Recommendation – We can target the promotion campaign in these cities and states where we can filter out the higher sales cities to run the promotional campaigns.

1. **In-depth Exploration:**  
   1. Is there a growing trend in the no. of orders placed over the past years?

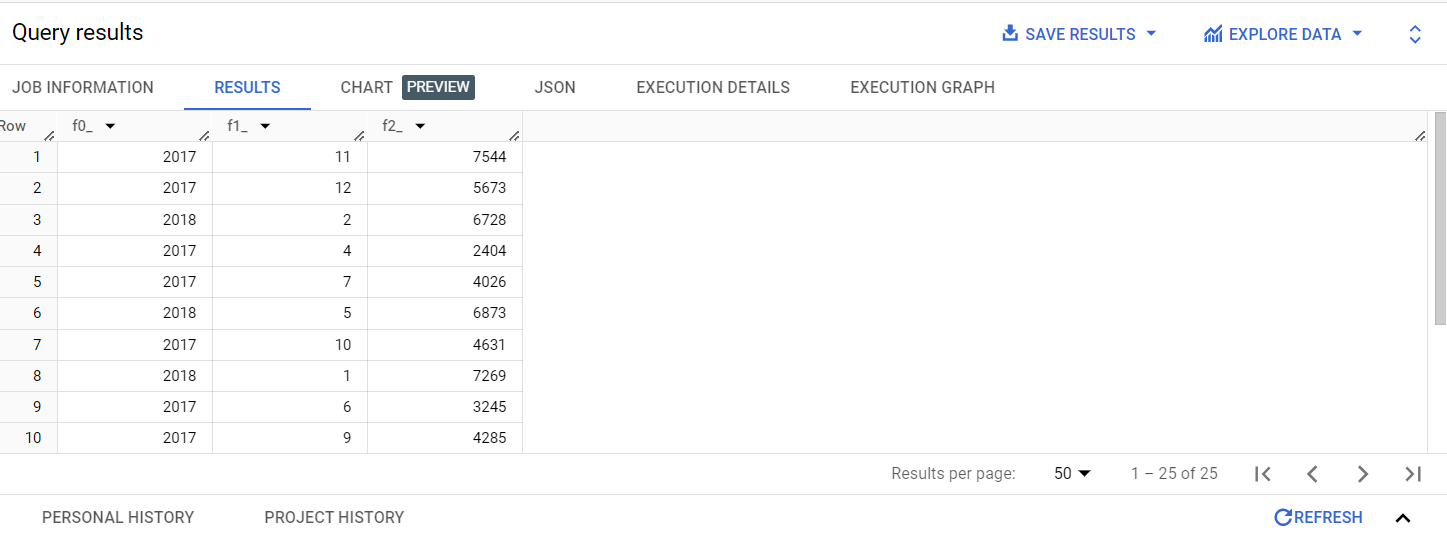
**SOLUTION -;** select extract(year from order\_purchase\_timestamp

),extract(month from order\_purchase\_timestamp

),count(order\_id) from target\_sql.orders

group by extract(year from order\_purchase\_timestamp

),extract(month from order\_purchase\_timestamp);

****

**Observation - ; my observation is that in September Month of 2016 sale is lowest in no. and where as in the November month of 2017 sales is highest and overall 2017 year sale is higher in comparison of the year 2016 and 2018.**

**Recommendation -: Promotions campaign need to be run in the month of November & December as we had a really low sales during these month in the 2 consecutive years (2016 & 2018).**

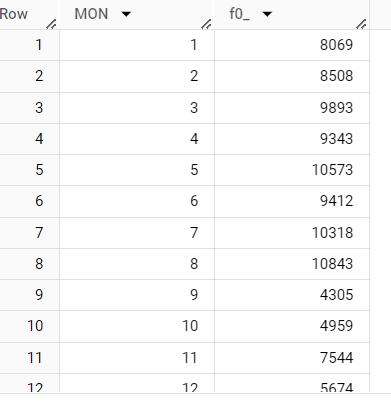
* 1. Can we see some kind of monthly seasonality in terms of the no. of orders being placed?

**SOLUTION –** select extract(month from order\_purchase\_timestamp

) AS MON,count(order\_id) from target\_sql.orders

group by MON

Order by Mon;



**Observation – I have observed that we can see the monthly sales is good in first & second quarter in each month but in the 3rd and 4th quarter specially in the month of November & December as we had a really low sales during these month in the 2 consecutive years (2016 & 2018).**

* 1. During what time of the day, do the Brazilian customers mostly place their orders? (Dawn, Morning, Afternoon or Night)
     1. 0-6 hrs : Dawn
     2. 7-12 hrs : Mornings
     3. 13-18 hrs : Afternoon
     4. 19-23 hrs : Night

SOLUTION - 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"

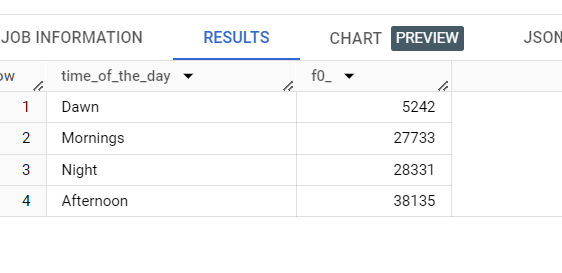
           end as time\_of\_the\_day,

           count(distinct customer\_id)

           from `target\_sql.orders`

           group by 1

           order by 2;



OBSERVATION – My observation is that mostly Brazilian people place the order during the afternoon time and least during the Dwan time.

1. **Evolution of E-commerce orders in the Brazil region:**
   1. Get the month on month no. of orders placed in each state.

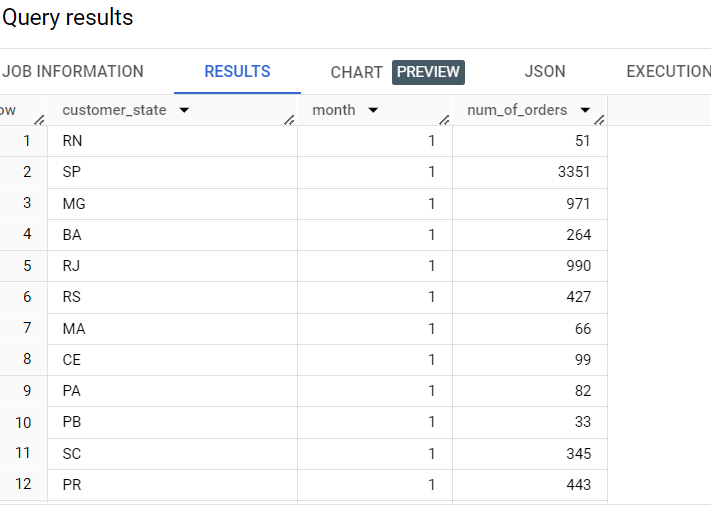
**SOLUTION -** select customer\_state,extract(month from order\_purchase\_timestamp

) as month,count(order\_id) num\_of\_orders from `target\_sql.customers` c

LEFT join `target\_sql.orders` o using(customer\_id)

group by customer\_state,month

order by month;



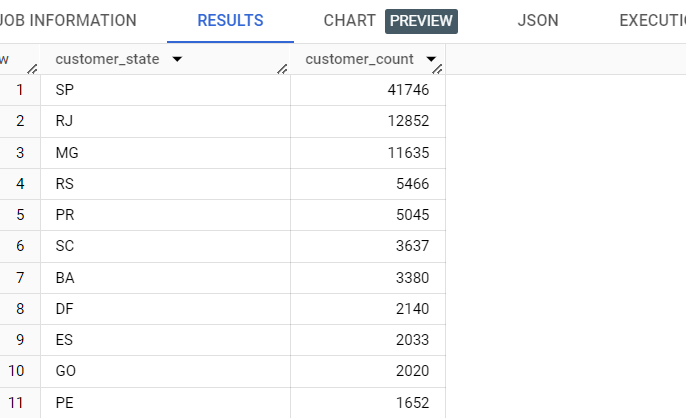
* 1. How are the customers distributed across all the states?

SOLUTION - SELECT customer\_state,count(distinct customer\_id) as customer\_count

 FROM `target\_sql.customers`

 group by customer\_state

 order by customer\_count desc;



OBSERVATION – MY Observation is that SP sate have the highest customer count and RJ state have the 2nd highest customer count but the difference huge in terms in no of customers count between the SP state and RJ state. RR have the lowest Customer count.

1. **Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.**
   1. Get the % increase in the cost of orders from year 2017 to 2018 (include months between Jan to Aug only).  
      You can use the "payment\_value" column in the payments table to get the cost of orders.

**SOLUTION -** with cte\_1 as (select extract (year from o.order\_purchase\_timestamp) as year,

sum(payment\_value) as pur\_1

from `target\_sql.orders` o inner join `target\_sql.payments` p using(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 year

),

cte\_2 as (select extract (year from o.order\_purchase\_timestamp) as year,

sum(payment\_value) as pur\_2

from `target\_sql.orders` o inner join `target\_sql.payments` p using(order\_id)

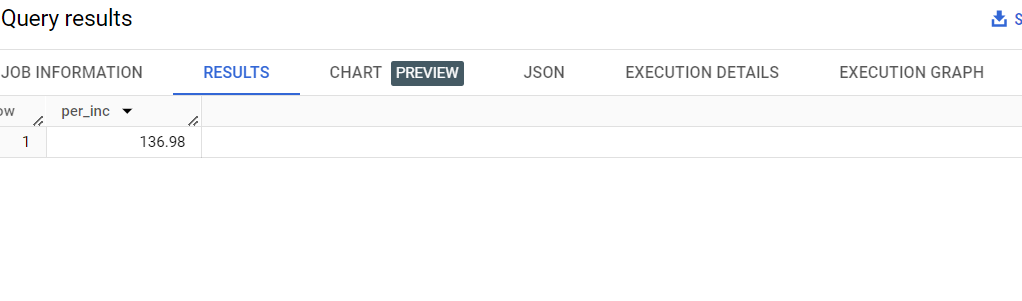
where extract(year from o.order\_purchase\_timestamp) =2018 and

      extract(month from o.order\_purchase\_timestamp) between 1 and 8

      group by year

)

select round((pur\_2-pur\_1)/pur\_1 \*100,2) as per\_inc from cte\_1,cte\_2;

****

**Observation – my observation is that 137% approx. price has increased from 2017 to 2018.**

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

**Solution -** select customer\_state,

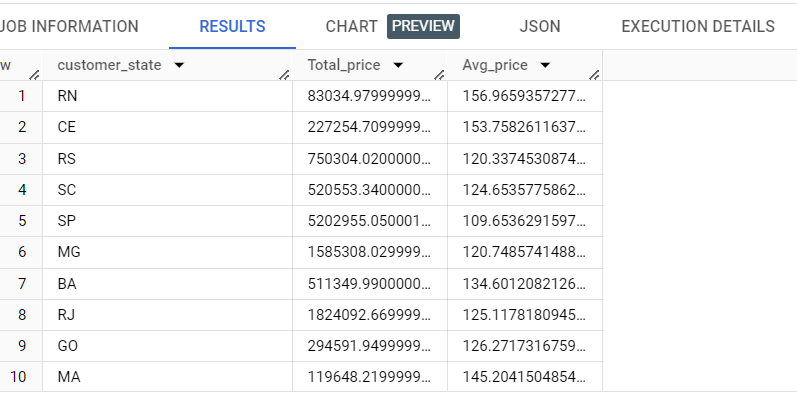
sum(price) as Total\_price,avg(price) as Avg\_price

 from `target\_sql.customers` c

left join `target\_sql.orders` o using(customer\_id)

left join `target\_sql.order\_items` oi using(order\_id)

group by customer\_state

****

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

**SOLUTION -** select customer\_state,

sum(freight\_value) as Total\_price,

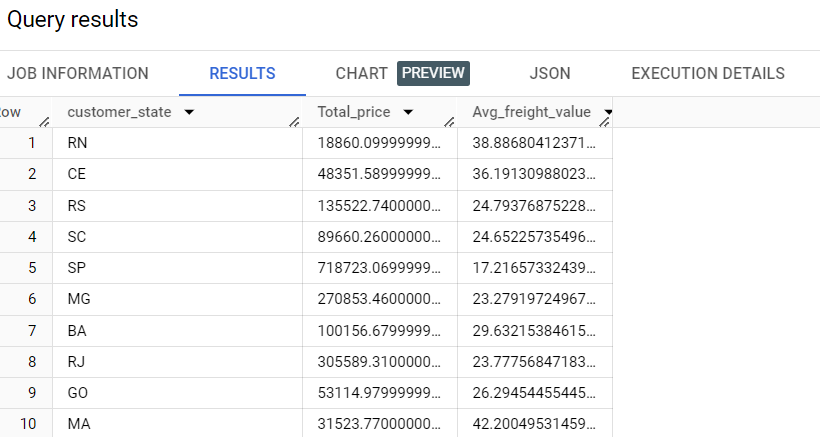
sum(freight\_value)/count(distinct order\_id) as Avg\_freight\_value

 from `target\_sql.customers` c

left join `target\_sql.orders` o using(customer\_id)

left join `target\_sql.order\_items` oi using(order\_id)

group by customer\_state

****

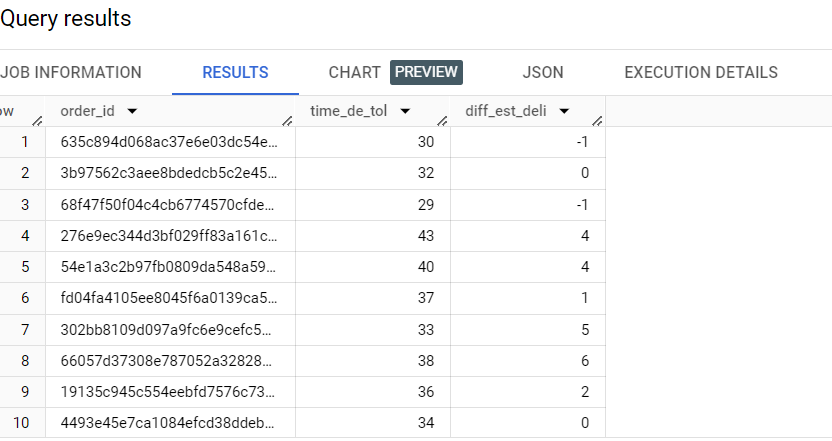
1. **Analysis based on sales, freight and delivery time.**
   1. Find the no. of days taken to deliver each order from the order’s purchase date as delivery time.  
      Also, calculate the difference (in days) between the estimated & actual delivery date of an order.  
      Do this in a single query.  
        
      You can calculate the delivery time and the difference between the estimated & actual delivery date using the given formula:
      1. **time\_to\_deliver** = order\_delivered\_customer\_date - order\_purchase\_timestamp
      2. **diff\_estimated\_delivery** = order\_estimated\_delivery\_date - order\_delivered\_customer\_date

**SOLUTION -** select order\_id, timestamp\_diff(order\_delivered\_customer\_date,order\_purchase\_timestamp,day) as time\_de\_tol,

                 timestamp\_diff(order\_delivered\_customer\_date,order\_estimated\_delivery\_date,day) as diff\_est\_deli

                 from `target\_sql.orders`

                 where order\_status = "delivered"

****

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

Solution – (Top 5 Highest states) - select customer\_state,

sum(freight\_value) as Total\_price,

sum(freight\_value)/count(distinct order\_id) as Avg\_freight\_value

 from `target\_sql.customers` c

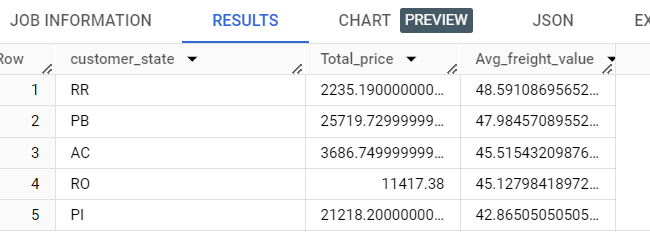
left join `target\_sql.orders` o using(customer\_id)

left join `target\_sql.order\_items` oi using(order\_id)

group by customer\_state

order by Avg\_freight\_value desc

limit 5;



solution – (Top 5 Lowest) - select customer\_state,

sum(freight\_value) as Total\_price,

sum(freight\_value)/count(distinct order\_id) as Avg\_freight\_value

 from `target\_sql.customers` c

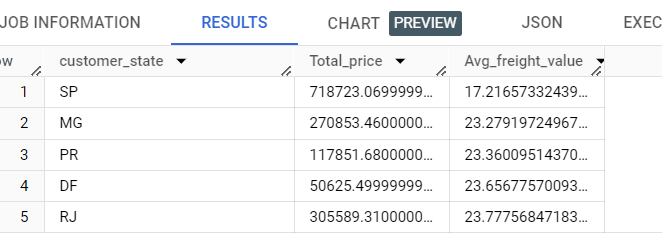
left join `target\_sql.orders` o using(customer\_id)

left join `target\_sql.order\_items` oi using(order\_id)

group by customer\_state

order by Avg\_freight\_value ASC

limit 5;



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

SOLUTION – (TOP 5 Highest Avg delivery time) –

select customer\_state,avg(timestamp\_diff(order\_delivered\_customer\_date,order\_purchase\_timestamp,day)) as time\_de\_tol

                 from `target\_sql.orders` o

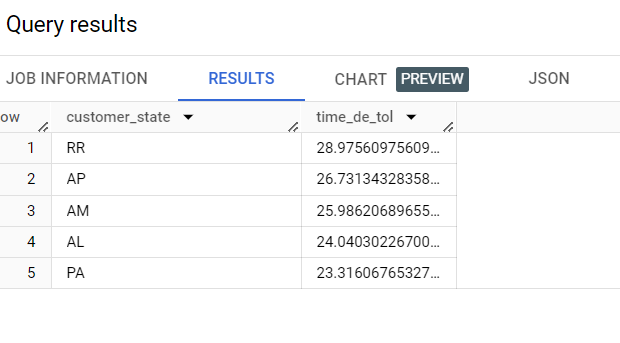
                 left join `target\_sql.customers` c using(customer\_id)

                 where order\_status = "delivered"

group by customer\_state

order by time\_de\_tol desc

limit 5;



Solution - (TOP 5 Lowest Avg delivery time) – select customer\_state,avg(timestamp\_diff(order\_delivered\_customer\_date,order\_purchase\_timestamp,day)) as time\_de\_tol

                 from `target\_sql.orders` o

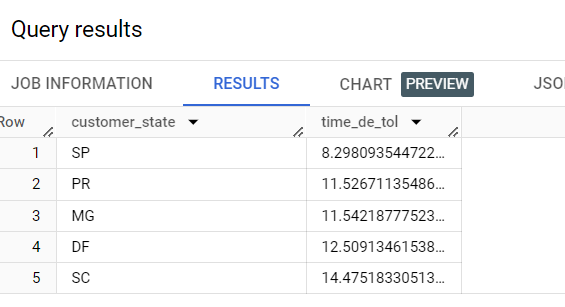
                 left join `target\_sql.customers` c using(customer\_id)

                 where order\_status = "delivered"

group by customer\_state

order by time\_de\_tol

limit 5;



* 1. 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

**Solution -** select customer\_state,

                 round(sum(timestamp\_diff(order\_delivered\_customer\_date,order\_purchase\_timestamp,day))/count(order\_id),2) as t1,

                 round(sum(timestamp\_diff(order\_delivered\_customer\_date,order\_estimated\_delivery\_date,day))/count(order\_id),2) as t2

                 from `target\_sql.orders`

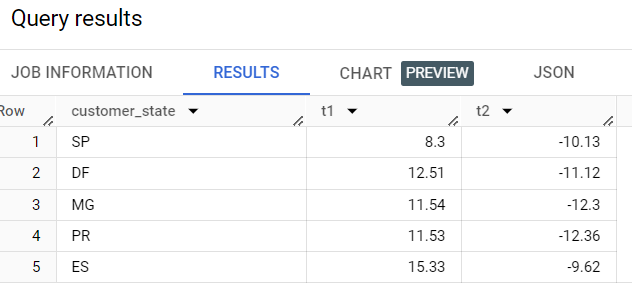
                 right join `target\_sql.customers` c using(customer\_id)

                 where order\_status = "delivered"

                 group by customer\_state

order by t1-t2

limit 5;

****

1. **Analysis based on the payments:**
   1. Find the month on month no. of orders placed using different payment types.

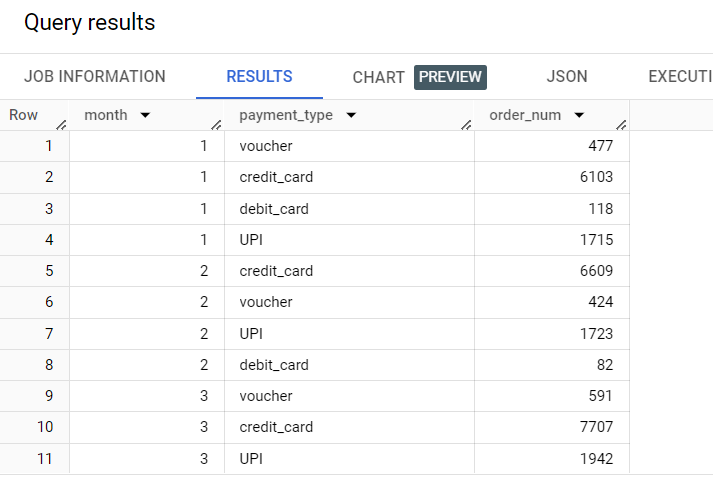
**SOLUTION -** select extract(month from order\_purchase\_timestamp) as month,payment\_type,

count(order\_id) as order\_num from `target\_sql.payments` p

 left join `target\_sql.orders` o using(order\_id)

group by month,payment\_type

order by month;

****

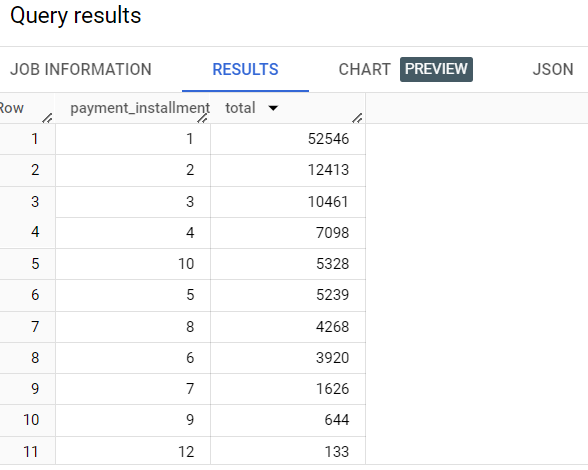
* 1. Find the no. of orders placed on the basis of the payment installments that have been paid.

**Solution -** select payment\_installments,count(order\_id) as total from `target\_sql.payments`

where payment\_installments >= 1

group by payment\_installments

order by total desc;

****

**OBSERVATION – I have observed that mostly orders are placed with the only 1 instalment means only 1 instalment received in majority of the orders and full instalment received in very less no of orders.**