****Q1 What does 'good' look like?****

****Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset:****

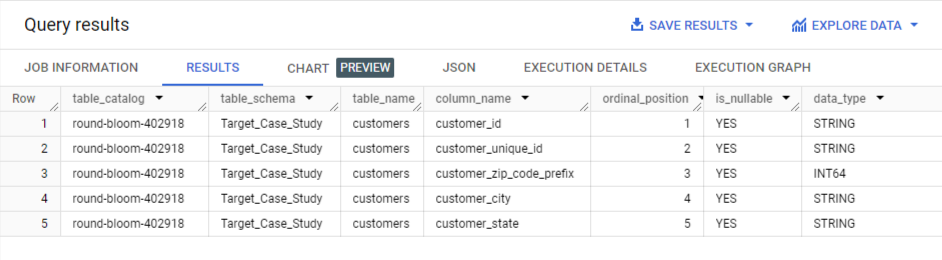
1. Data type of all columns in the "customers" table.

Query :

select \* from round-bloom-402918.Target\_Case\_Study.INFORMATION\_SCHEMA.COLUMNS

where TABLE\_NAME='customers'

Output :

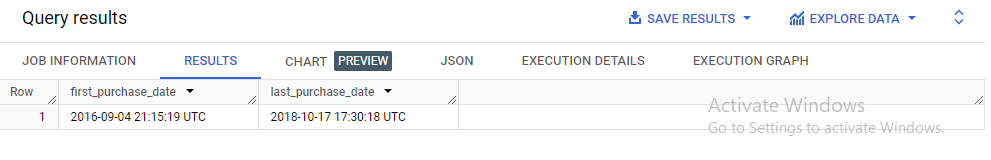


2. Get the time range between which the orders were placed.

Query :

select min(order\_purchase\_timestamp) as first\_purchase\_date, max(order\_purchase\_timestamp) as last\_purchase\_date from round-bloom-402918.Target\_Case\_Study.orders

Output :



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

Query :

select count(distinct c.customer\_city) as no\_of\_cities, count(distinct c.customer\_state) as no\_of\_states

from round-bloom-402918.Target\_Case\_Study.orders o

join round-bloom-402918.Target\_Case\_Study.customers c

on o.customer\_id=c.customer\_id

where o.order\_purchase\_timestamp between '2016-09-04 21:15:19 UTC' and '2018-10-17 17:30:18 UTC'

Output :



****2. In-depth Exploration:****

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

Query :

select distinct extract(year from order\_purchase\_timestamp) as year,

count(order\_id) over(partition by extract(year from order\_purchase\_timestamp)) as no\_of\_orders\_this\_year

from round-bloom-402918.Target\_Case\_Study.orders

order by year

Output :

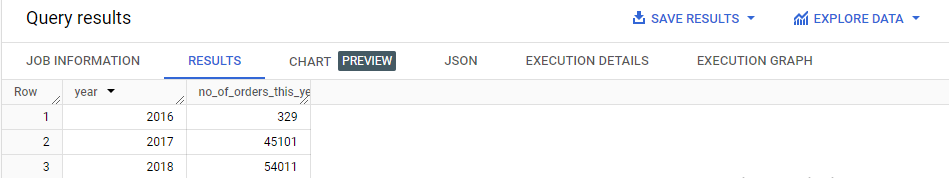
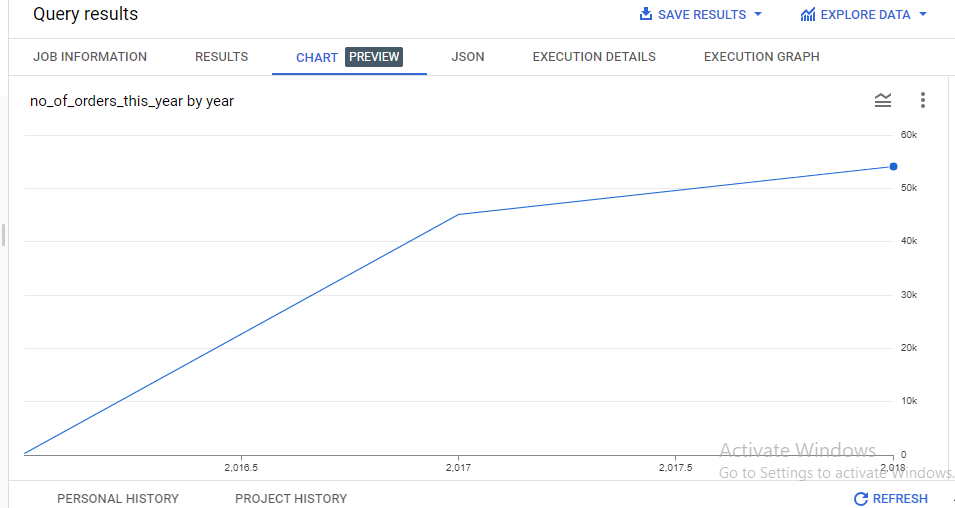


Chart :



Insight :

Yes, there is a growing trend as can be seen from numbers and steady upward trajectory of the line chart.

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

Query :

select format\_date('%m', order\_purchase\_timestamp) as Month,

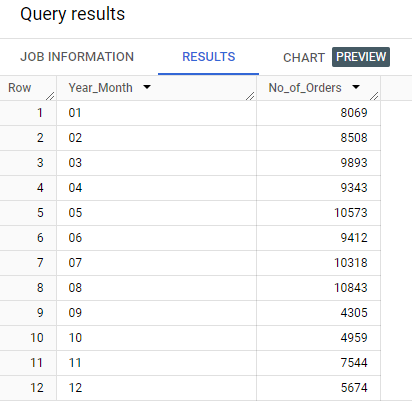
count(order\_id) as No\_of\_Orders

from round-bloom-402918.Target\_Case\_Study.orders

group by Month

order by Month

Output :

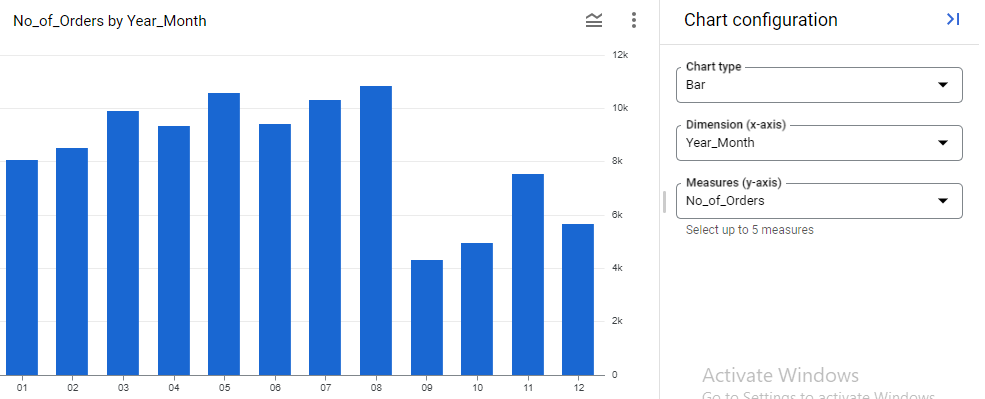


Insights :

As we can see from the bar graph the number of orders keeps picking up in the first half of the year and then a significant drop happens after the month of August where the orders are at peak levels. Orders drop consecutively in next two months of Sept and Oct by a huge margin from peak values.

Nov and Dec show some recovery.

Chart :



3. 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\_of\_day,

count(distinct order\_id) as No\_of\_Orders

from round-bloom-402918.Target\_Case\_Study.orders

group by time\_of\_day

order by No\_of\_Orders desc

Output :

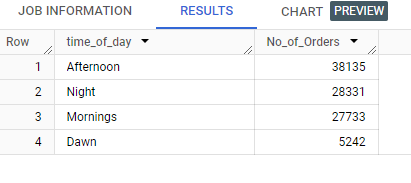
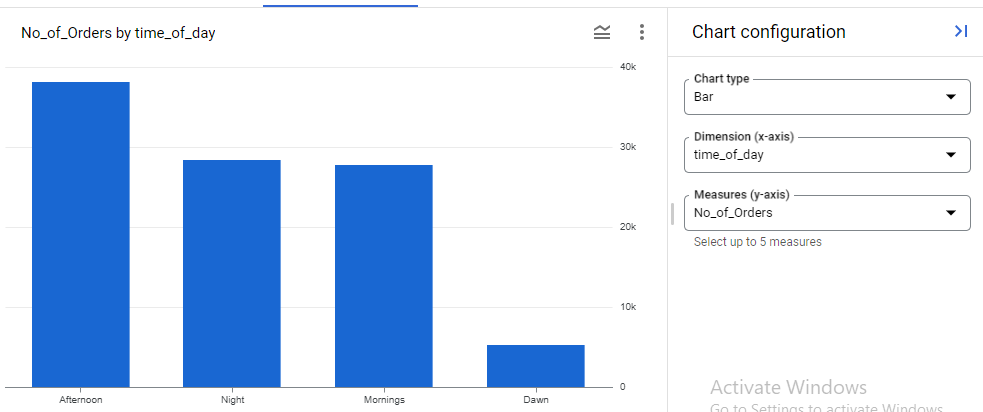


Chart :



Insights:

As we can clearly see from the numbers and the bar graph that most orders were placed by the Brazilian customers during the afternoon period(13-18 hrs) and least during Dawn(0-6 hrs).

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

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

Query :

WITH MonthlyOrders AS (

  SELECT

    FORMAT\_TIMESTAMP('%Y-%m', o.order\_purchase\_timestamp) AS monthyear,

    c.customer\_state,

    COUNT(o.order\_id) AS No\_of\_Orders

  FROM

    round-bloom-402918.Target\_Case\_Study.orders o

    join round-bloom-402918.Target\_Case\_Study.customers c

    on o.customer\_id=c.customer\_id

  GROUP BY

    monthyear, c.customer\_state

)

SELECT

  monthyear,

  customer\_state,

  No\_of\_Orders,

  LEAD(No\_of\_Orders) OVER (PARTITION BY customer\_state ORDER BY monthyear) AS Nxt\_Month\_Orders,

  SAFE\_DIVIDE(LEAD(No\_of\_Orders) OVER (PARTITION BY customer\_state ORDER BY monthyear) - No\_of\_Orders, No\_of\_Orders) \* 100 AS Month\_on\_Month\_pct

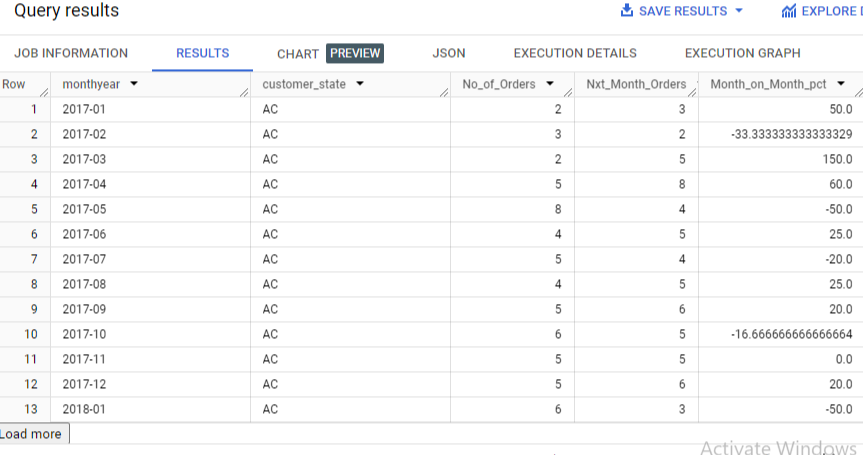
FROM

  MonthlyOrders

ORDER BY

  customer\_state, monthyear;

Output :



Insights :

Shows state-wise monthly orders and next-month’s orders to calculate month-on-month growth or decline in the sales. Negative indicates decline in sales and positive month-on-month % indicates growth in sales.

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

Query :

select c.customer\_state, count(c.customer\_id) as customers\_in\_state

from round-bloom-402918.Target\_Case\_Study.customers c

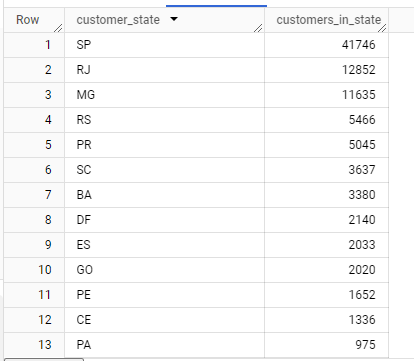
join round-bloom-402918.Target\_Case\_Study.orders o

on c.customer\_id=o.customer\_id

group by c.customer\_state

order by customers\_in\_state desc

Output :



Insights:

SP has highest number of customers with 41746 and RR has the lowest with 46.

****4. 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.

Query :

with cte1 as(

select sum(p.payment\_value) as cost\_of\_orders\_2017, extract(year from o.order\_purchase\_timestamp) as previous\_year

from round-bloom-402918.Target\_Case\_Study.payments p

join round-bloom-402918.Target\_Case\_Study.orders o

on p.order\_id=o.order\_id

where extract(month from order\_purchase\_timestamp) between 01 and 08

and extract(year from order\_purchase\_timestamp) = 2017

group by extract(year from order\_purchase\_timestamp)

),

cte2 as(

select sum(p.payment\_value) as cost\_of\_orders\_2018, extract(year from order\_purchase\_timestamp) as current\_year

from round-bloom-402918.Target\_Case\_Study.payments p

join round-bloom-402918.Target\_Case\_Study.orders o

on p.order\_id=o.order\_id

where extract(month from order\_purchase\_timestamp) between 01 and 08

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

group by extract(year from order\_purchase\_timestamp)

)

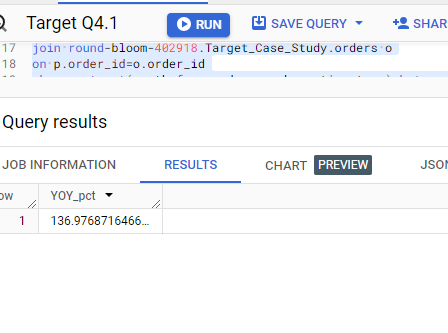
select ((cte2.cost\_of\_orders\_2018 - cte1.cost\_of\_orders\_2017)/cte1.cost\_of\_orders\_2017)\*100 as YOY\_pct

from cte1

join cte2

on cte1.previous\_year+1=cte2.current\_year

Output :



Insights:

There is a growth in cost of orders from 2017 of approx. 137% in the year 2018.

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

Query :

with cte1 as (

select

c.customer\_state as state,

sum(price) as total\_price,

count(distinct o.order\_id) as No\_of\_Orders

from round-bloom-402918.Target\_Case\_Study.customers c

left join round-bloom-402918.Target\_Case\_Study.orders o

on c.customer\_id=o.customer\_id

left join round-bloom-402918.Target\_Case\_Study.order\_items oi

on o.order\_id=oi.order\_id

group by state

)

select cte1.state,

cte1.total\_price,

(cte1.total\_price/cte1.No\_of\_Orders) as avg\_price

from cte1

order by cte1.total\_price desc

Output :

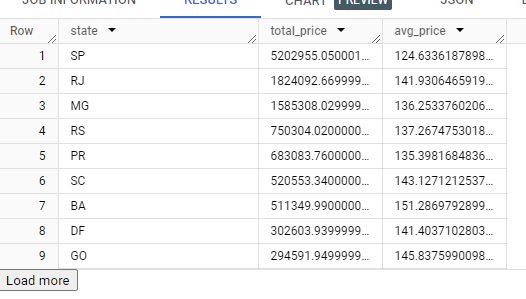
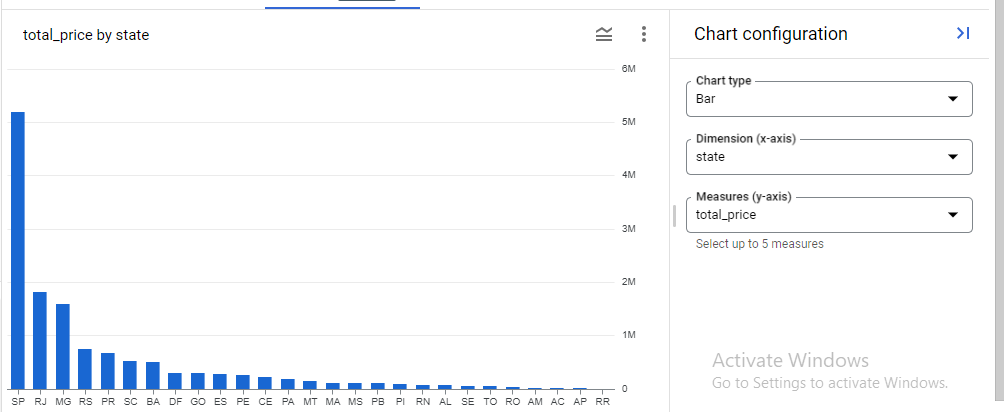
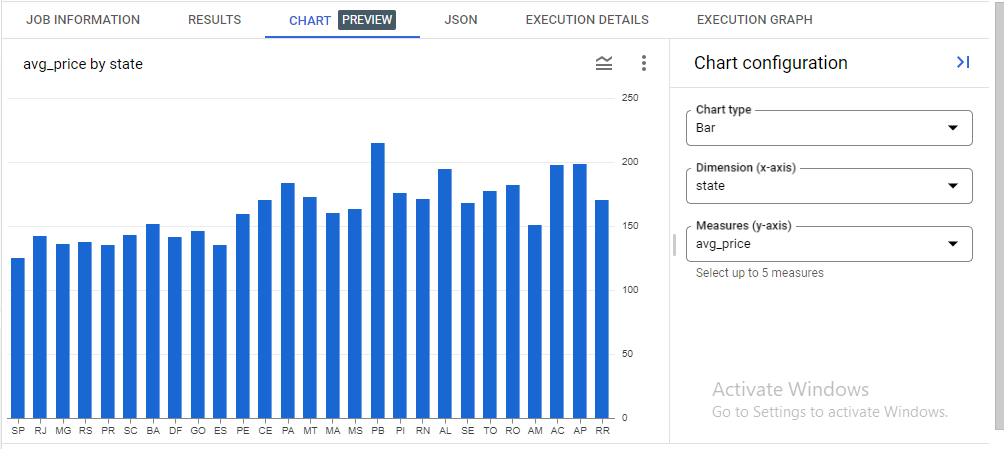


Chart :





Insights :

SP has the highest total order price at 5202955.05 and RR has the lowest total at 7829.43.

But SP has the lowest average order price at 124.63 while PB has the highest average at 215.05.

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

Query :

with cte1 as (

select

c.customer\_state as state,

sum(freight\_value) as total\_freight\_value,

count(distinct o.order\_id) as No\_of\_Orders

from round-bloom-402918.Target\_Case\_Study.customers c

left join round-bloom-402918.Target\_Case\_Study.orders o

on c.customer\_id=o.customer\_id

left join round-bloom-402918.Target\_Case\_Study.order\_items oi

on o.order\_id=oi.order\_id

group by state

)

select cte1.state,

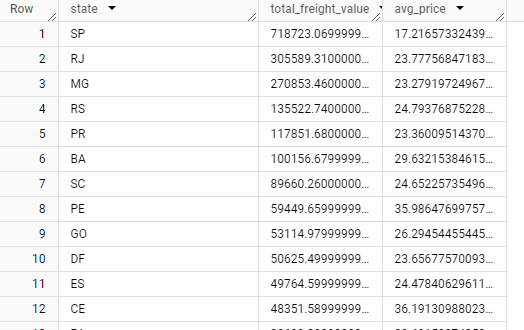
cte1.total\_freight\_value,

(cte1.total\_freight\_value/cte1.No\_of\_Orders) as avg\_price

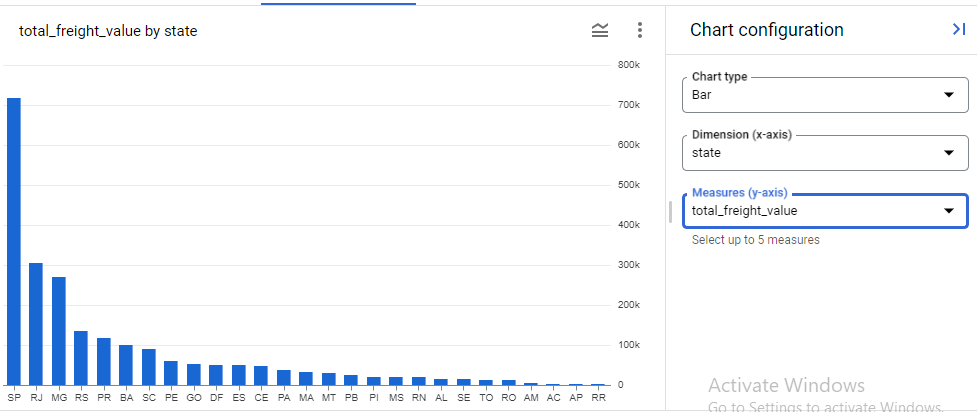
from cte1

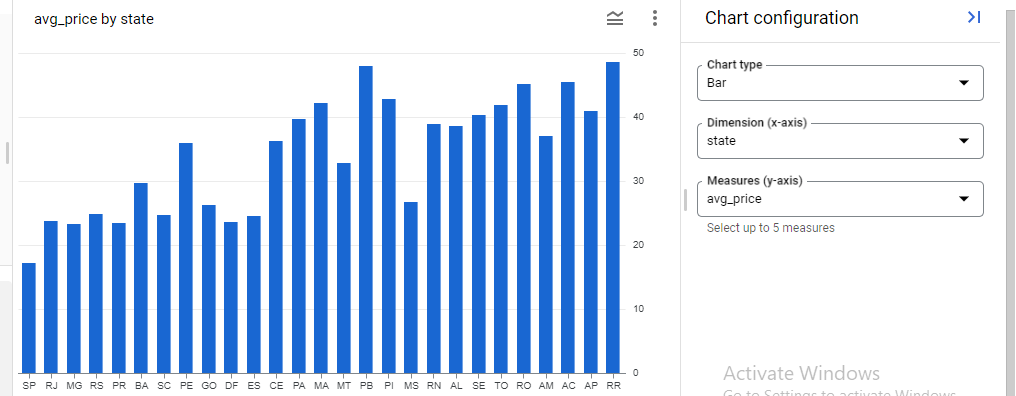
order by cte1.total\_freight\_value desc

Output :



Charts :





Insights:

SP has the highest total freight value at 718723.07 and RR has the lowest total at 2235.19.

But RR has the highest average freight value at 48.591 while SP has the lowest average at 17.21.

****5. 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.

Query :

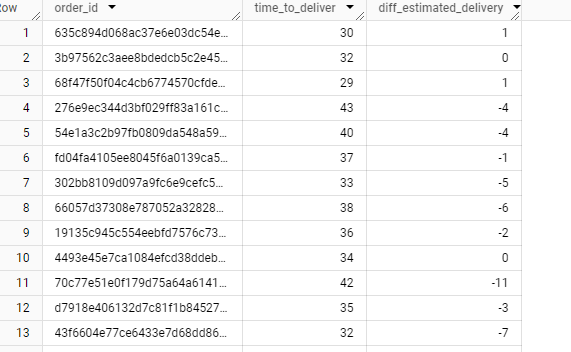
select order\_id, timestamp\_diff( order\_delivered\_customer\_date, order\_purchase\_timestamp, day) as time\_to\_deliver,

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

from round-bloom-402918.Target\_Case\_Study.orders

where order\_status = 'delivered'

Output :



Insights:

The time\_to\_deliver mentions time taken for order to be delivered to the customer’s place as we can see in this output, mostly around the month mark. Diff\_estimated\_delivery mentions the number of days by the orders deviated from expected delivery dates where the negative values indicate late delivery.

2. i) Find out the top 5 states with the highest average freight value.

Query:

select c.customer\_state, round(((sum(oi.freight\_value))/(count(distinct o.order\_id))),2) as avg\_freight\_per\_state

from round-bloom-402918.Target\_Case\_Study.order\_items oi

join round-bloom-402918.Target\_Case\_Study.orders o

on oi.order\_id=o.order\_id

join round-bloom-402918.Target\_Case\_Study.customers c

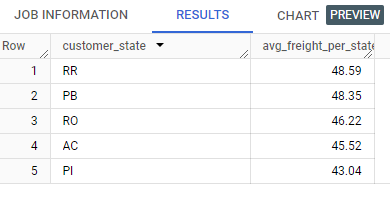
on c.customer\_id=o.customer\_id

group by c.customer\_state

order by avg\_freight\_per\_state desc

limit 5

Output:



Insights:

RR, PB, RO, AC and PI are the top 5 states with the highest average freight values upwards of 43.

2. ii) Find out the top 5 states with the lowest average freight value.

Query:

select c.customer\_state, round(((sum(oi.freight\_value))/(count(distinct o.order\_id))),2) as avg\_freight\_per\_state

from round-bloom-402918.Target\_Case\_Study.order\_items oi

join round-bloom-402918.Target\_Case\_Study.orders o

on oi.order\_id=o.order\_id

join round-bloom-402918.Target\_Case\_Study.customers c

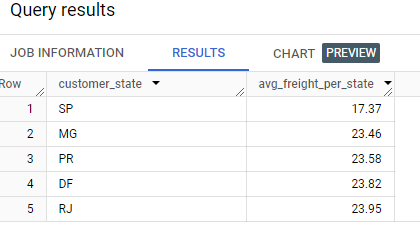
on c.customer\_id=o.customer\_id

group by c.customer\_state

order by avg\_freight\_per\_state

limit 5

Output:



Insights:

SP, MG, PR, DF and RJ are the states with the lowest average freight values in the range of 17 to 24.

3. i) Find out the top 5 states with the highest average delivery time.

Query:

select t.customer\_state, sum(t.delivery\_time)/count(distinct t.order\_id) as average\_delivery\_time

from

( select date\_diff( o.order\_delivered\_customer\_date, o.order\_purchase\_timestamp, day) as delivery\_time,

c.customer\_state, o.order\_id

from round-bloom-402918.Target\_Case\_Study.orders o

join round-bloom-402918.Target\_Case\_Study.customers c

on c.customer\_id=o.customer\_id

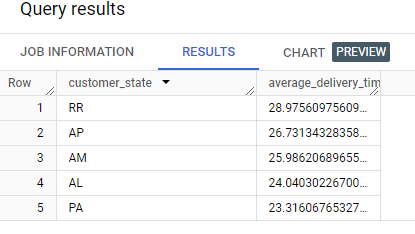
where o.order\_status='delivered')t

group by t.customer\_state

order by average\_delivery\_time desc

limit 5

Output:



Insights:

RR, AP, AM, AL and PA are the states with the highest average delivery time which is in the range of 23 to 29 days.

3. ii) Find out the top 5 states with the lowest average delivery time.

Query:

select t.customer\_state, sum(t.delivery\_time)/count(distinct t.order\_id) as average\_delivery\_time

from

( select date\_diff( o.order\_delivered\_customer\_date, o.order\_purchase\_timestamp, day) as delivery\_time,

c.customer\_state, o.order\_id

from round-bloom-402918.Target\_Case\_Study.orders o

join round-bloom-402918.Target\_Case\_Study.customers c

on c.customer\_id=o.customer\_id

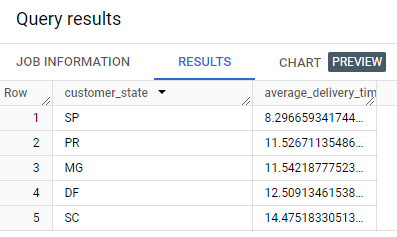
where o.order\_status='delivered')t

group by t.customer\_state

order by average\_delivery\_time

limit 5

Output:



Insights:

SP, PR, MG, DF and SC are the states with the lowest average delivery time which is in the range of 8 to 15 days.

4. 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:

with cte1 as (

select t.customer\_state, sum(t.delivery\_time)/count(distinct t.order\_id) as average\_delivery\_time

from

( select date\_diff( o.order\_delivered\_customer\_date, o.order\_purchase\_timestamp, day) as delivery\_time,

c.customer\_state, o.order\_id

from round-bloom-402918.Target\_Case\_Study.orders o

join round-bloom-402918.Target\_Case\_Study.customers c

on c.customer\_id=o.customer\_id

where o.order\_status='delivered')t

group by t.customer\_state

order by average\_delivery\_time),

cte2 as

(select t.customer\_state, sum(t.estimated\_delivery\_time)/count(distinct t.order\_id) as average\_estimated\_delivery\_time

from

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

c.customer\_state, o.order\_id

from round-bloom-402918.Target\_Case\_Study.orders o

join round-bloom-402918.Target\_Case\_Study.customers c

on c.customer\_id=o.customer\_id)t

group by t.customer\_state

order by average\_estimated\_delivery\_time)

select cte1.customer\_state, (cte2.average\_estimated\_delivery\_time - cte1.average\_delivery\_time) as fastest\_delivery\_time

from

cte1

join

cte2

on cte1.customer\_state=cte2.customer\_state

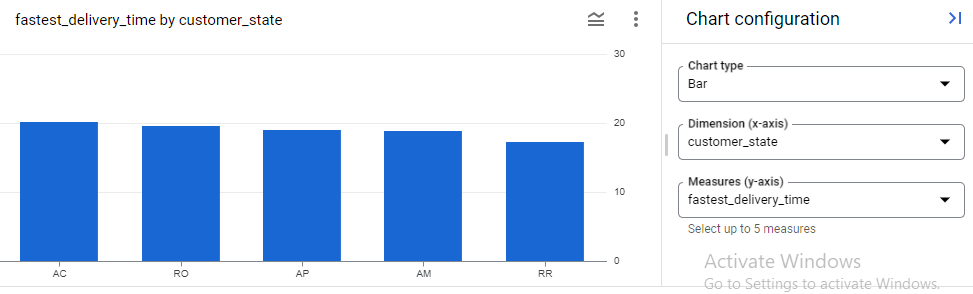
order by fastest\_delivery\_time desc

limit 5

Output :



Chart:



Insights:

AC, RO, AP, AM and RR are the states with the fastest delivery times as compared to the estimated date of delivery.

1. ****Analysis based on the payments:****

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

Query:

select t.monthyear,

t.payment\_type,

count(t.payment\_type) as no\_of\_orders,

lead(t.monthyear) over(partition by t.payment\_type order by t.monthyear) as next\_monthyear,

lead(count(t.payment\_type),1) over(partition by t.payment\_type order by t.monthyear) as next\_month\_no\_of\_orders,

round(safe\_divide(lead(count(t.payment\_type),1) over(partition by

t.payment\_type order by t.monthyear) - count(t.payment\_type), count(t.payment\_type)),2) \* 100 as Month\_on\_Month\_pct

from

(select

row\_number() over (partition by format\_timestamp('%Y-%m',order\_purchase\_timestamp)) as row\_number,

format\_timestamp('%Y-%m',order\_purchase\_timestamp) as monthyear,

payment\_type

from round-bloom-402918.Target\_Case\_Study.payments p

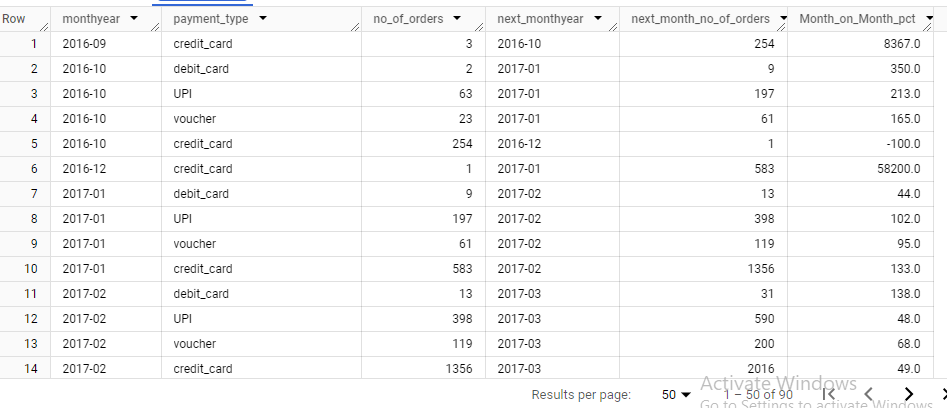
join round-bloom-402918.Target\_Case\_Study.orders o

on p.order\_id=o.order\_id)t

group by t.monthyear, t.payment\_type

order by t.monthyear

Output:



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

Query:

select payment\_installments, count(p.order\_id) as no\_of\_orders

from round-bloom-402918.Target\_Case\_Study.payments p

where payment\_installments >=1

and exists (

    select 1

    from round-bloom-402918.Target\_Case\_Study.orders o

    where o.order\_id = p.order\_id

  )

Output:

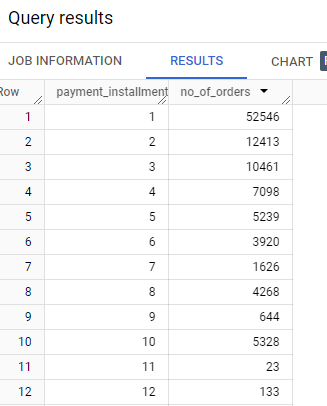


Chart:

