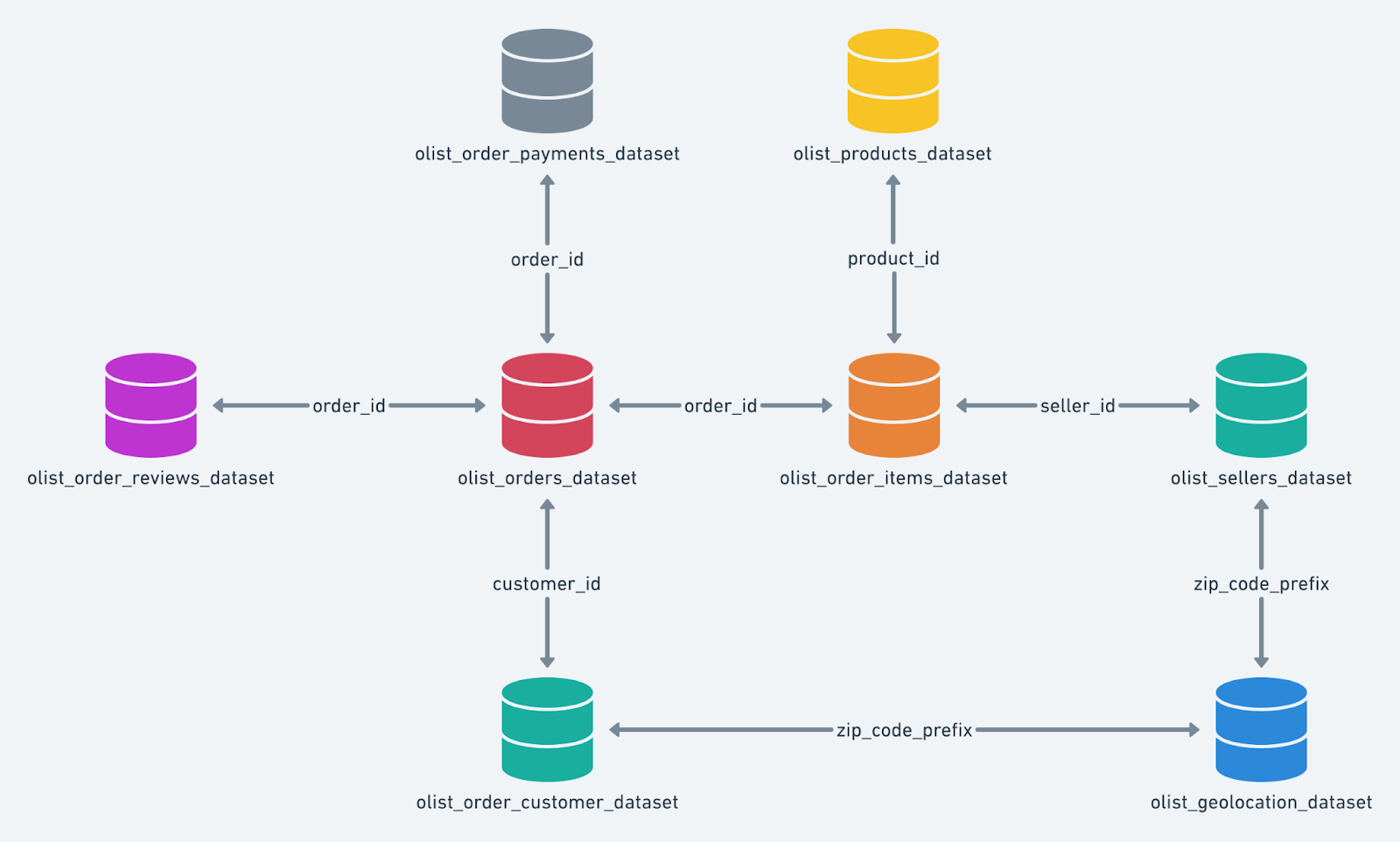
**Context:**

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, customer demographics, product characteristics, and customer satisfaction levels.



**Problem Statement:**

Analyze the given dataset to extract valuable insights and provide actionable recommendations.

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

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.
   2. Get the time range between which the orders were placed.
   3. Count the Cities & States of customers who ordered during the given period.
2. **In-depth Exploration:**
   1. Is there a growing trend in the no. of orders placed over the past years?
   2. Can we see some kind of monthly seasonality in terms of the no. of orders being placed?
   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
3. **Evolution of E-commerce orders in the Brazil region:**
   1. Get the month on month no. of orders placed in each state.
   2. How are the customers distributed across all the states?
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.
   2. Calculate the Total & Average value of order price for each state.
   3. Calculate the Total & Average value of order freight for each state.
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.  
        
      You can calculate the delivery time and the difference between the estimated & actual delivery date using the given formula:
      * **time\_to\_deliver** = order\_delivered\_customer\_date - order\_purchase\_timestamp
      * **diff\_estimated\_delivery** = order\_estimated\_delivery\_date - order\_delivered\_customer\_date
   2. Find out the top 5 states with the highest & lowest average freight value.
   3. Find out the top 5 states with the highest & lowest average delivery time.
   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.
6. **Analysis based on the payments:**
   1. Find the month on month no. of orders placed using different payment types.
   2. Find the no. of orders placed on the basis of the payment installments that have been paid.

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

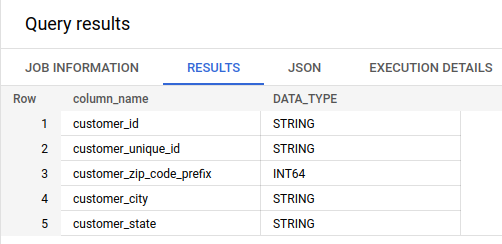
**1. Data type of columns in a table**

SELECT column\_name,DATA\_TYPE

FROM `target-sql-382405`.Target\_Dataset.INFORMATION\_SCHEMA.COLUMNS

WHERE

TABLE\_NAME = 'customers'

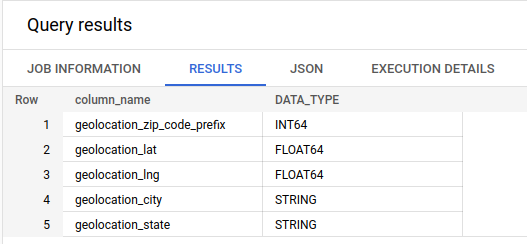


SELECT column\_name,DATA\_TYPE

FROM `target-sql-382405`.Target\_Dataset.INFORMATION\_SCHEMA.COLUMNS

WHERE

TABLE\_NAME = 'geolocation'

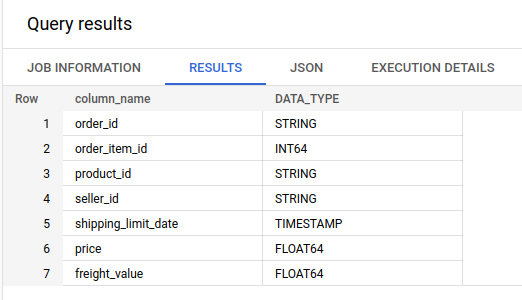


SELECT column\_name,DATA\_TYPE

FROM `target-sql-382405`.Target\_Dataset.INFORMATION\_SCHEMA.COLUMNS

WHERE

TABLE\_NAME = 'order\_items'

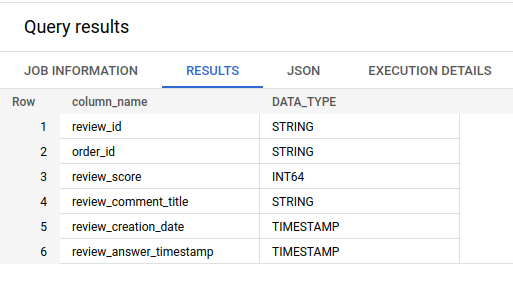


SELECT column\_name,DATA\_TYPE

FROM `target-sql-382405`.Target\_Dataset.INFORMATION\_SCHEMA.COLUMNS

WHERE

TABLE\_NAME = 'order\_reviews'

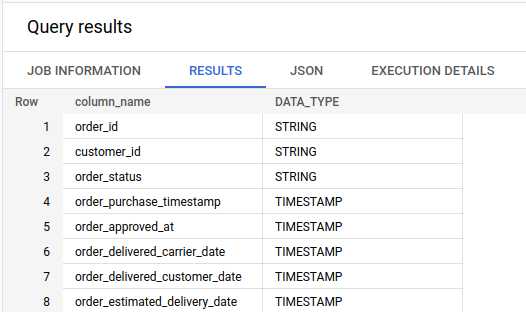


SELECT column\_name,DATA\_TYPE

FROM `target-sql-382405`.Target\_Dataset.INFORMATION\_SCHEMA.COLUMNS

WHERE

TABLE\_NAME = 'orders'

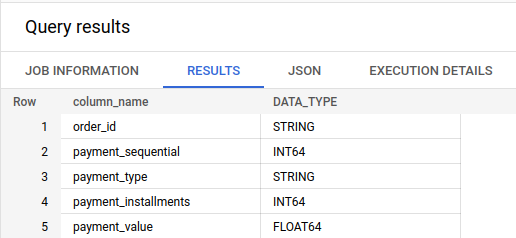


SELECT column\_name,DATA\_TYPE

FROM `target-sql-382405`.Target\_Dataset.INFORMATION\_SCHEMA.COLUMNS

WHERE

TABLE\_NAME = 'payments'

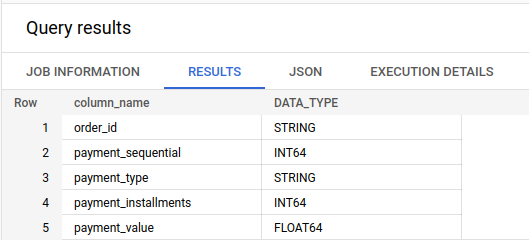


SELECT column\_name,DATA\_TYPE

FROM `target-sql-382405`.Target\_Dataset.INFORMATION\_SCHEMA.COLUMNS

WHERE

TABLE\_NAME = 'products'

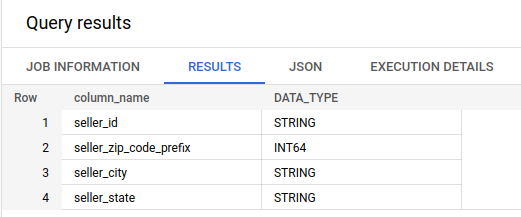


SELECT column\_name,DATA\_TYPE

FROM `target-sql-382405`.Target\_Dataset.INFORMATION\_SCHEMA.COLUMNS

WHERE

TABLE\_NAME = 'sellers'

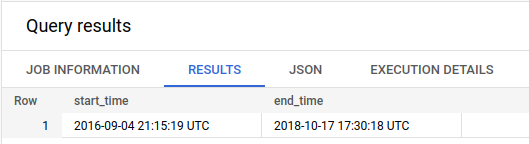


**2. Time period for which the data is given:**

SELECT min(order\_purchase\_timestamp) as start\_time, max(order\_purchase\_timestamp) as end\_time

FROM `target-sql-382405.Target\_Dataset.orders`

Here I have considered the time period of order\_purchase\_timestamp since there is nothing mentioned specifically in the question.



**3. City and states of customers ordered during the given time period:**

SELECT count(c.customer\_id) as customer\_count,customer\_city,customer\_state

FROM `target-sql-382405.Target\_Dataset.customers` as c

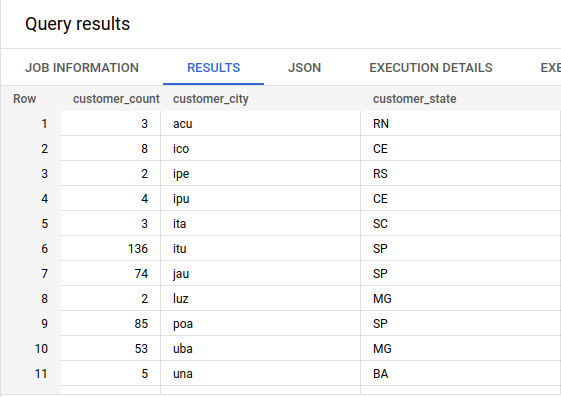
join `Target\_Dataset.orders` as o on c.customer\_id=o.customer\_id

where order\_purchase\_timestamp between (SELECT min(order\_purchase\_timestamp) FROM `target-sql-382405.Target\_Dataset.orders` ) and (select max(order\_purchase\_timestamp) FROM `target-sql-382405.Target\_Dataset.orders` )

group by customer\_state,customer\_city

**Assumptions:**

Here the time period is considered based on the order\_purchase\_timestamp.



**2. 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?**

SELECT count(customer\_id) as customer\_count,

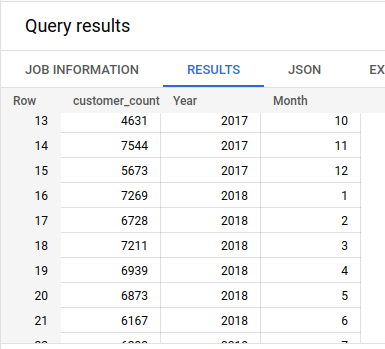
extract(year FROM order\_purchase\_timestamp) as Year,

extract(month FROM order\_purchase\_timestamp) as Month

FROM `target-sql-382405.Target\_Dataset.orders`

group by Year,Month

order by Year,Month



From the year 2016 month of september till 2018 month of october we have the data. Among that we could see the count of customer increase month over month. From 11th month of 2017 to 3rd month of 2018 there is some increase in between for the months november 2017,January 2018,march 2018. So the seasonality could fall under these months. Mostly late fall and winter season.

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

SELECT count(customer\_id) as customer\_count,

case

when extract(hour FROM order\_purchase\_timestamp) between 3 and 6 then 'Dawn'

when extract(hour FROM order\_purchase\_timestamp) between 7 and 12 then 'Morning'

when extract(hour FROM order\_purchase\_timestamp) between 13 and 18 then 'Afternoon'

else 'Night'

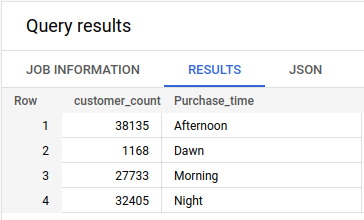
end as Purchase\_time

from `Target\_Dataset.orders` group by Purchase\_time order by Purchase\_time

Assumptions:

I assume that the order\_purchase\_timestamp is in brazilian timestamp and based on that calculated the time customers tend to buy.

Orders are counted irrespective of delivered or undelivered, cancelled or paid partially or completely



From the above result, it is observed that there is more customer traffic in the afternoon, that is between 13 hrs and 18 hrs. ‘Night’ also is have second less customer traffic.

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

**1.** Get month on month orders by states

SELECT \*,

ROUND((z.order\_id\_count - (LAG(z.order\_id\_count,1) OVER(PARTITION BY z.customer\_state ORDER BY z.customer\_state, Year, Month)))\*100/LAG(z.order\_id\_count,1) OVER(PARTITION BY z.customer\_state ORDER BY z.customer\_state, Year, Month),2) AS m\_o\_m\_per100

FROM

(SELECT DISTINCT c.customer\_state,COUNT(o.order\_id) AS order\_id\_count,

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

FROM `Target\_Dataset.orders` AS o

JOIN `Target\_Dataset.customers` AS c

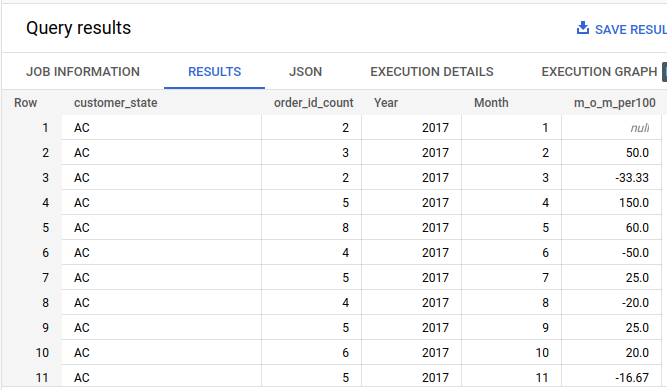
USING(customer\_id)

GROUP BY c.customer\_state, Year, Month

ORDER BY c.customer\_state, Year, Month) AS z

ORDER BY z.customer\_state, Year, Month

Not exactly, there is a growing trend in orders as it initially (2016) and then later (2018) we see a sharp dip in orders. Initially we see 2017 March. We get to see a peak in 2017 March. Also, we can find peaks during 2017 Nov and 2018 Jan, March. With given data we find peak in March repeated, indicating seasonality in the region.



**2. Distribution of customers across the states in Brazil**

SELECT v.customer\_state,concat(ROUND(v.c\_count\*100/SUM(v.c\_count) OVER(),2),'%') AS percentage\_distribution

FROM(

SELECT DISTINCT COUNT(customer\_id) AS c\_count, customer\_state

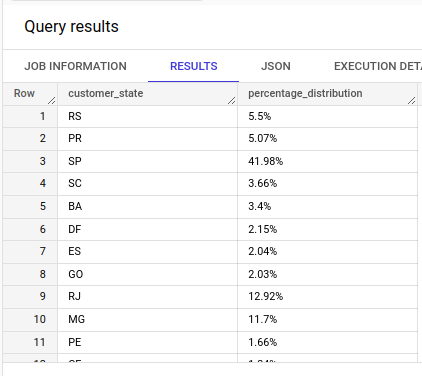
FROM `Target\_Dataset.customers`

GROUP BY customer\_state) AS v

ORDER BY percentage\_distribution DESC

Assumptions and insights:

From the result obtained, it is found that the maximum customer distribution is from the state named “**SP**” .



**4. Impact on Economy: Analyze 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) - You can use “payment\_value” column in payments table**

Assumptions and insights:

\* But when we do see rise in payment value between 2016 to 2017 only considering Jan to Aug month on month even with oscillating growth percentages

SELECT \*,

CONCAT(ROUND((x.tot\_pay\_val - (LAG(x.tot\_pay\_val,1) OVER(ORDER BY year, month)))\*100/LAG(x.tot\_pay\_val,1) OVER(ORDER BY year, month),2),"%") AS m\_o\_m\_per100

FROM

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

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

ROUND(SUM(p.payment\_value),2) AS tot\_pay\_val,

FROM `Target\_Dataset.orders` AS o

JOIN `Target\_Dataset.payments` AS p

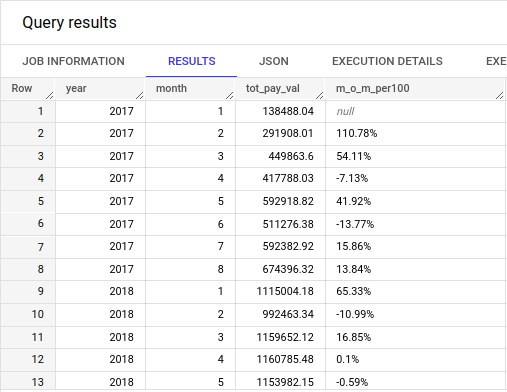
ON o.order\_id = p.order\_id

GROUP BY year,month

ORDER BY year,month) AS x

WHERE x.year BETWEEN 2017 AND 2018 AND x.month BETWEEN 1 AND 8

ORDER BY year, month



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

select

customer\_state,

sum(price) as Sum\_price\_per\_state,

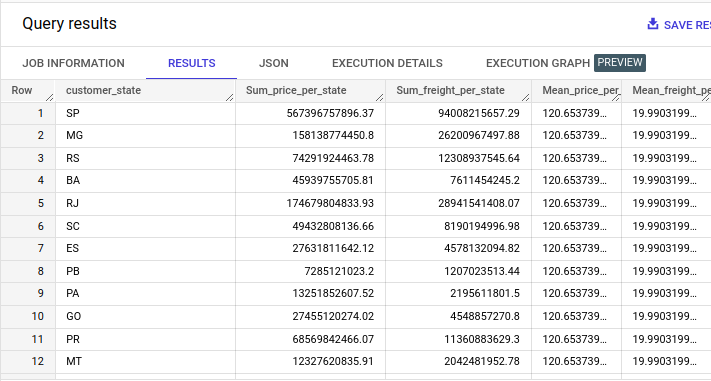
sum(freight\_value) as Sum\_freight\_per\_state,

avg(price) as Mean\_price\_per\_state,

avg(freight\_value) as Mean\_freight\_per\_state

from `Target\_Dataset.order\_items`,`Target\_Dataset.customers`

group by customer\_state



**5. Analysis on sales, freight and delivery time**

1. **Calculate days between purchasing, delivering and estimated delivery**

select

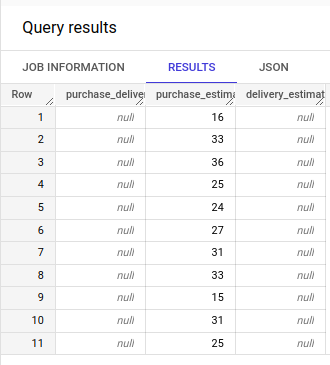
date\_diff(order\_purchase\_timestamp, order\_delivered\_customer\_date, day) as purchase\_delivery\_timeperiod,

date\_diff(order\_estimated\_delivery\_date,order\_purchase\_timestamp, day) as purchase\_estimated\_delivery,

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

from `Target\_Dataset.orders`

order by purchase\_delivery\_timeperiod



**2. Find time\_to\_delivery & diff\_estimated\_delivery. Formula for the same given below:**

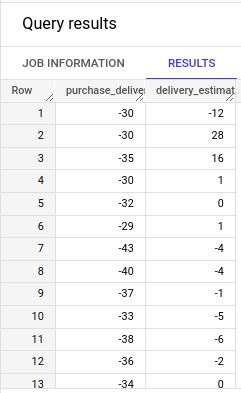
* **time\_to\_delivery = order\_purchase\_timestamp-order\_delivered\_customer\_date**
* **diff\_estimated\_delivery = order\_estimated\_delivery\_date-order\_delivered\_customer\_date**

select

date\_diff(order\_purchase\_timestamp, order\_delivered\_customer\_date, day) as purchase\_delivery\_timeperiod,

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

from `Target\_Dataset.orders`



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

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

* Top 5 states with highest average freight value - sort in desc/asc limit 5

select

customer\_state,

avg(ot.freight\_value) average\_freight

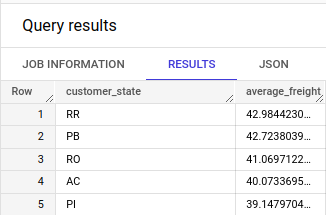
from `Target\_Dataset.order\_items` as ot

join `Target\_Dataset.orders` as o on ot.order\_id=o.order\_id

join `Target\_Dataset.customers` as c on o.customer\_id=c.customer\_id

group by c.customer\_state

order by average\_freight desc



Top 5 states with highest average time to delivery

select

customer\_state,

avg(DATE\_DIFF(order\_delivered\_customer\_date,order\_purchase\_timestamp,day)) as average\_delivery

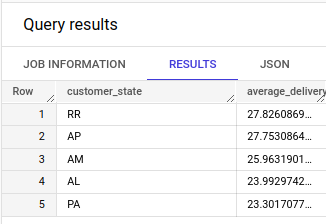
from `Target\_Dataset.order\_items` as ot

join `Target\_Dataset.orders` as o on ot.order\_id=o.order\_id

join `Target\_Dataset.customers` as c on o.customer\_id=c.customer\_id

group by c.customer\_state

order by average\_delivery desc limit 5



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

**select \*,**

(average\_estimated\_delivery-average\_delivery) as delivery\_difference

from

(

select

customer\_state,

avg(DATE\_DIFF(order\_delivered\_customer\_date,order\_purchase\_timestamp,day)) as average\_delivery,

avg(DATE\_DIFF(order\_estimated\_delivery\_date,order\_purchase\_timestamp,day)) as average\_estimated\_delivery

from `Target\_Dataset.order\_items` as ot

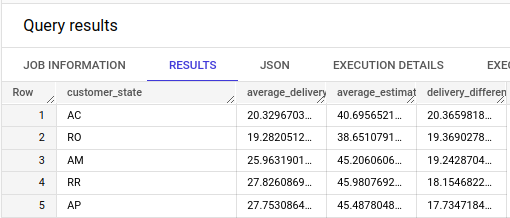
join `Target\_Dataset.orders` as o on ot.order\_id=o.order\_id

join `Target\_Dataset.customers` as c on o.customer\_id=c.customer\_id

group by c.customer\_state

)

order by delivery\_difference desc limit 5

****

**6. Payment type analysis:**

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

select \*,

round(((x.order\_count-lag(x.order\_count) over(partition by payment\_type order by Year,Month))\*100)/lag(x.order\_count) over(partition by payment\_type order by Year,Month),2) as Month\_on\_Month\_orders

from

(

select

count(o.order\_id) as order\_count,

payment\_type,

extract(month from order\_purchase\_timestamp) as Month,

extract(year from order\_purchase\_timestamp) as Year

from `Target\_Dataset.orders` as o join `Target\_Dataset.payments` as p

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

group by payment\_type,Year,Month

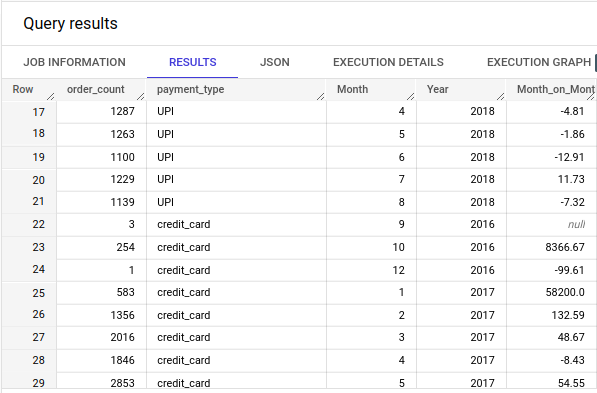
order by payment\_type,Year,Month

) as x

Assumptions and Insights:

- We see customers prefer credit card payments more than UPI and debit card.

- For those customers who pay by installments, we see high number for less than 10 installments and few for more than 10 installments



1. **Count of orders based on the no. of payment installments**

SELECT p.payment\_installments, COUNT(p.order\_id) AS orders\_count

FROM `SQL\_Target\_Project.payments` AS p

JOIN `SQL\_Target\_Project.orders` AS o

USING(order\_id)

GROUP BY p.payment\_installments;

