TARGET - BUSINESS CASE STUDY

## 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.**

ANS:-

SELECT \*

FROM `sql-1-406201.Target\_sql.INFORMATION\_SCHEMA.COLUMNS`

WHERE Table\_name = 'customers';

**OUTPUT:-**



**INSIGHTS**:-

* In this output ,there are five rows in that four rows are “String” Data type except customer\_zip\_code\_prefix which is “INT64” .

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

ANS:-

SELECT min(order\_purchase\_timestamp) AS `min`

,MAX(order\_purchase\_timestamp) AS `max` FROM `Target\_sql.orders`

**OUTPUT:-**



**INSIGHTS**:-

* The first order placed in 2016-09-04 21:15:19 UTC
* The last order placed in 2018-10-17 17:30:18 UTC

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

ANS:-

SELECT count(DISTINCT C.customer\_city) AS CITY,count(DISTINCT C.customer\_state) AS STATES

FROM `Target\_sql.customers` AS C INNER JOIN `Target\_sql.orders` AS O ON C.customer\_id = O.customer\_id

WHERE O.order\_purchase\_timestamp BETWEEN "2016-01-01" AND "2018-12-

31";

# OUTPUT:-



**INSIGHTS:-**

* People from Brazil order from 4119 cities and 27 states during 2016-2018.

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

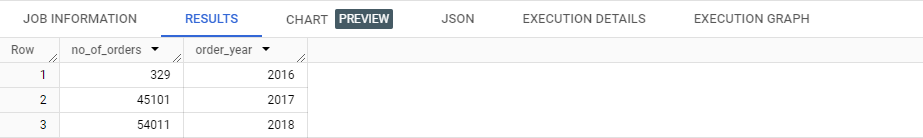
1. **Is there a growing trend in the no. of orders placed over the past years?**

ANS:-

SELECT COUNT(order\_id) AS `no\_of\_orders`,EXTRACT (YEAR FROM order\_purchase\_timestamp) AS `order\_year`,

FROM `Target\_sql.orders` GROUP BY order\_year ORDER BY order\_year ASC;

# OUTPUT:-



**INSIGHTS:-**

* + There is the gradual increased number of orders from 2016-2018.
  + Comparing the all three years in year 2016 there are only 329 orders which is less than 1000.
  + In 2017, there are 45101 orders which very high compare to the year 2016.
  + In 2018, there are 54011 orders where almost increased near to 10000 orders compare to 2017.

**RECOMMENDATIONS:-**

* + In 2016 , If Target runs any form of discounts and marketing through advertisements, social media, and digital marketing there may be chance of increasing the number of orders.

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

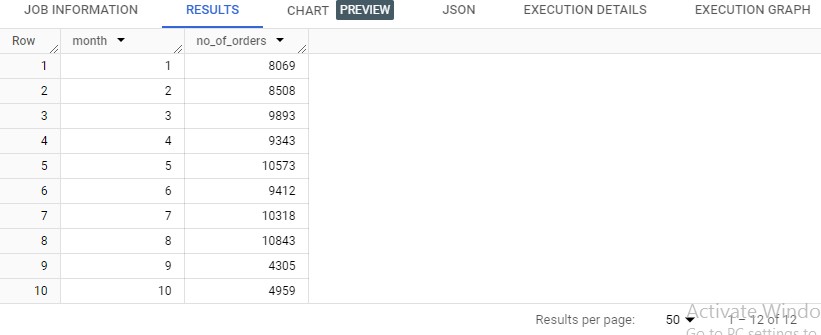
ANS:-

SELECT EXTRACT(MONTH FROM order\_purchase\_timestamp) AS `month`, COUNT(order\_id) AS `no\_of\_orders`

FROM `Target\_sql.orders` GROUP BY month

ORDER BY month;

# OUTPUT:-



**INSIGHTS:-**

* In the month of May , July and August there are high in sales compared to other months which are 10000 above orders.
* The highest number of orders placed in the month of August i.e. 10843 orders when compared to other months.
* The lowest number of orders placed in the month of September and October i.e. 4305 and 4959 orders which is before the August.
* In the month of January and February orders there are only 439 difference January – 8069

February – 8508

* In the month of March, April and June orders there is a decent amount of orders placed.

March – 9893

April – 9343

June - 9412

* In the month of May , July and August there are high in sales compared to other months which are 10000 above orders.
* There was a medium sales in the month of November and December

**Recommendations:-**

* The highest and lowest sales happened in followed by months only on observing that if we run any campaign or discounts at that particular time there might be chance of increasing sales till medium level.
* December is the Christimas Eve month many of them show interest in buying new things like Electronic appliances . Making this an advantage we should give 0 down payment and Emi options and people may buy new electronics .
* And also give the discount for the products which are for long period in the store by giving clearance sale in the month of December.

## 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

ANS:-

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 'Morning'

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'

ELSE 'Unknown' END AS time\_of\_day,

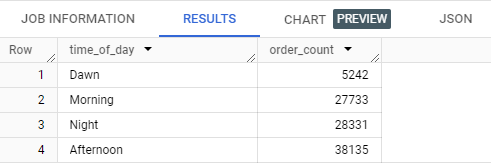
COUNT(order\_id) AS order\_count FROM

`Target\_sql.orders` GROUP BY

time\_of\_day ORDER BY

order\_count;

# OUTPUT:-



**INSIGHTS:-**

* There is a low order count in the dawn and high order count in Aftenoon.
* There is a Medium order count in Morning and Night when compared to Night.

**Recommendations:-**

* Put a timer discount for the premium members. So, that they can buy more orders at dawn.

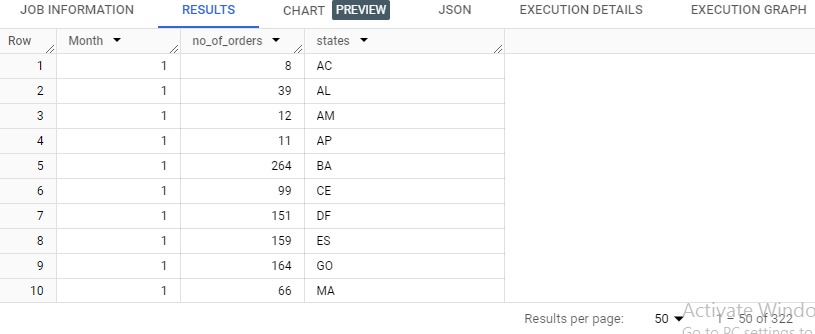
1. **Evolution of E-commerce orders in the Brazil region:**
   1. **Get the month on month no. of orders placed in each state.**

ANS:-

SELECT EXTRACT (MONTH FROM order\_purchase\_timestamp) AS `Month` , COUNT(O.order\_id) AS `no\_of\_orders` , C.customer\_state AS `states` FROM `Target\_sql.orders` AS O LEFT JOIN `Target\_sql.customers` AS C ON O.customer\_id = C.customer\_id

GROUP BY Month, states ORDER BY Month,states ;

# OUTPUT:-



**INSIGHTS:-**

* SP state has high number of orders in all the months.
* RR state has low number of orders in the 1st month.

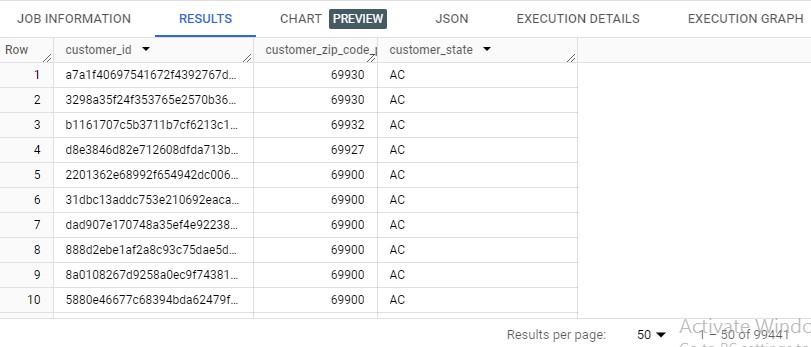
## How are the customers distributed across all the states?

ANS:-

SELECT customer\_id , customer\_zip\_code\_prefix, customer\_state FROM `Target\_sql.customers`

ORDER by customer\_state;

# OUTPUT:-



**INSIGHTS:-**

* There are 99441 rows and extracted the orders from each zip code from Brazil.

## 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.

ANS:- SELECT

((SUM(CASE WHEN EXTRACT(YEAR FROM order\_purchase\_timestamp ) = 2018 AND EXTRACT(MONTH FROM order\_purchase\_timestamp) BETWEEN 1 AND

8 THEN payment\_value ELSE 0 END) -

SUM(CASE WHEN EXTRACT(YEAR FROM order\_purchase\_timestamp) = 2017 AND EXTRACT(MONTH FROM order\_purchase\_timestamp) BETWEEN 1 AND

8 THEN payment\_value ELSE 0 END))

/

SUM(CASE WHEN EXTRACT(YEAR FROM order\_purchase\_timestamp) = 2017 AND EXTRACT(MONTH FROM order\_purchase\_timestamp) BETWEEN 1 AND

8 THEN payment\_value ELSE 0 END)

) \* 100 AS percentage\_increase

FROM `Target\_sql.payments` AS P INNER JOIN `Target\_sql.orders` AS O ON P.order\_id = O.order\_id

WHERE EXTRACT(YEAR FROM order\_purchase\_timestamp) IN (2017, 2018);

# OUTPUT:-



**INSIGHTS:-**

* 136.9768716466 percentage increased in the cost of orders from year 2017 to 2018 between January to August only.

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

ANS:-

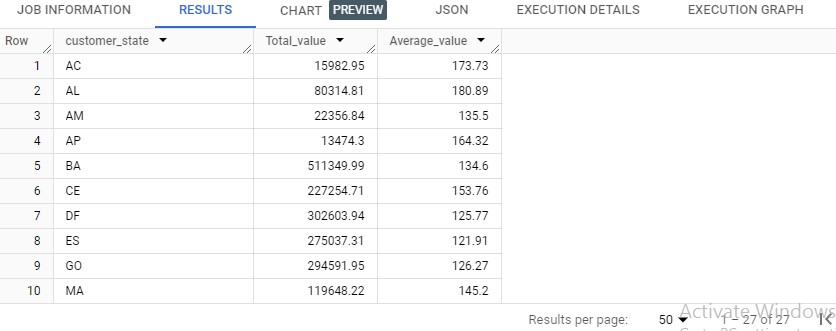
SELECT C.customer\_state, ROUND (SUM(OD.price),2) AS `Total\_value`, ROUND (AVG(OD.price),2) AS `Average\_value`,

FROM `Target\_sql.customers` AS C INNER JOIN `Target\_sql.orders` AS O ON C.customer\_id = O.customer\_id

INNER JOIN `Target\_sql.order\_items` AS OD ON O.order\_id = OD.order\_id

GROUP BY C.customer\_state ORDER BY C.customer\_state;

# OUTPUT:-



**INSIGHTS:-**

* There are total 27 rows present in this output.
* SP state has the highest total value of comparing all other states.
* RR state has the lowest total value of comparing all other states.
* PB state has the highest average value of comparing all other states.
* SP state has the lowest average value of comparing all other states.

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

**ANS:-**

SELECT C.customer\_state, ROUND (SUM(freight\_value),2) AS

`Total\_Freight\_value`, ROUND (AVG(freight\_value),2) AS

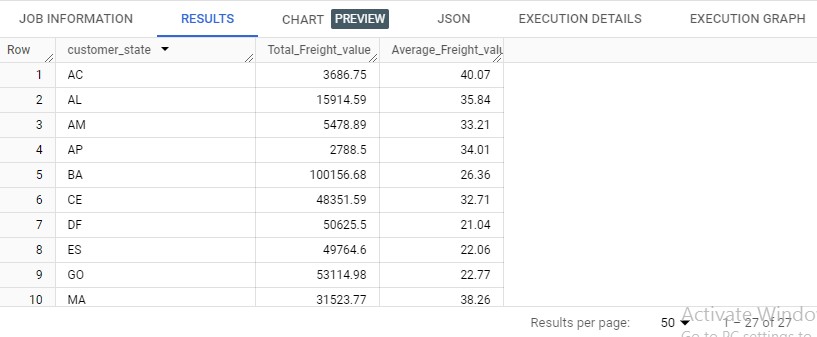
`Average\_Freight\_value`,

FROM `Target\_sql.customers` AS C INNER JOIN `Target\_sql.orders` AS O ON C.customer\_id = O.customer\_id

INNER JOIN `Target\_sql.order\_items` AS OD ON O.order\_id = OD.order\_id

GROUP BY C.customer\_state ORDER BY C.customer\_state;

# OUTPUT:-



**INSIGHTS:-**

* There are total 27 rows present in this output.
* RJ state has the highest total frieght value of comparing all other states.
* AP state has the lowest total freight value of comparing all other states.
* RR state has the highest average freight value of comparing all other states.
* SP state has the lowest average freight value of comparing all other states.

## 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**

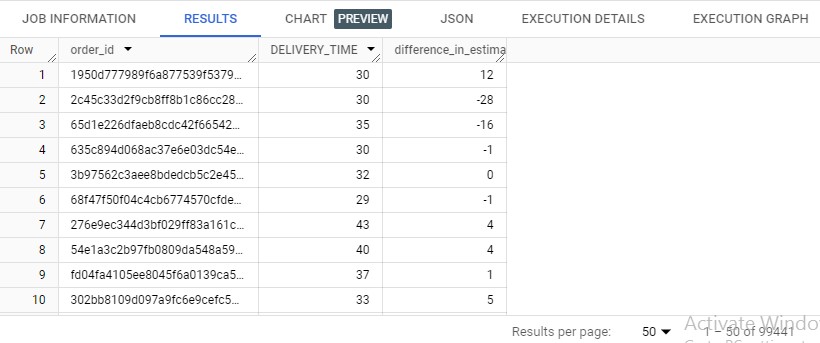
## diff\_estimated\_delivery = order\_delivered\_customer\_date - order\_estimated\_delivery\_date

ANS:-

SELECT order\_id, DATE\_DIFF(order\_delivered\_customer\_date , order\_purchase\_timestamp,DAY) AS `DELIVERY\_TIME` , DATE\_DIFF (order\_delivered\_customer\_date , order\_estimated\_delivery\_date,DAY) AS

`difference\_in\_estimated\_delivery` FROM `Target\_sql.orders`

# OUTPUT:-



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

WITH Maxm AS (

SELECT C.customer\_state AS `AVG\_MAX\_STATES`, ROUND (AVG(D.freight\_value)) AS `AVG\_MAX\_OF\_FREIGHT\_VALUES`

FROM `Target\_sql.customers` AS C INNER JOIN `Target\_sql.orders` AS O ON C.customer\_id = O.Customer\_id

INNER JOIN `Target\_sql.order\_items` AS D ON O.order\_id = D.order\_id

GROUP BY AVG\_MAX\_STATES

ORDER BY AVG\_MAX\_OF\_FREIGHT\_VALUES DESC

)

Sa AS (

SELECT C.customer\_state AS `AVG\_Min\_STATES`, ROUND (AVG(D.freight\_value)) AS `AVG\_MIN\_OF\_FREIGHT\_VALUES`

FROM `Target\_sql.customers` AS C INNER JOIN `Target\_sql.orders` AS O ON C.customer\_id = O.Customer\_id

INNER JOIN `Target\_sql.order\_items` AS D ON O.order\_id = D.order\_id

GROUP BY AVG\_Min\_STATES

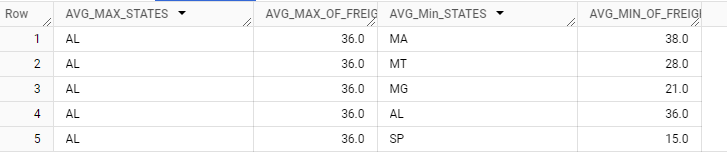
ORDER BY AVG\_MIN\_OF\_FREIGHT\_VALUES ASC

)

SELECT AVG\_MAX\_STATES, AVG\_MAX\_OF\_FREIGHT\_VALUES, AVG\_Min\_STATES, AVG\_MIN\_OF\_FREIGHT\_VALUES

FROM Maxm, Sa LIMIT 5;

# OUTPUT:-



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

WITH HIGH AS(

SELECT C.customer\_state AS `H\_STATE`, ROUND(AVG(DATE\_DIFF(order\_delivered\_customer\_date,order\_purchase\_tim estamp,DAY)),2) AS `HIGHEST\_AVERAGE\_DELIVERY`

FROM `Target\_sql.customers` AS C INNER JOIN `Target\_sql.orders` AS O ON C.customer\_id = O.customer\_id

GROUP BY C.customer\_state ORDER BY C.customer\_state DESC

),

LOW AS(

SELECT C.customer\_state AS

`L\_STATE`,ROUND(AVG(DATE\_DIFF(order\_delivered\_customer\_date,order\_pu rchase\_timestamp,DAY)),2) AS `LOWEST\_AVERAGE\_DELIVERY`

FROM `Target\_sql.customers` AS C INNER JOIN `Target\_sql.orders` AS O ON C.customer\_id = O.customer\_id

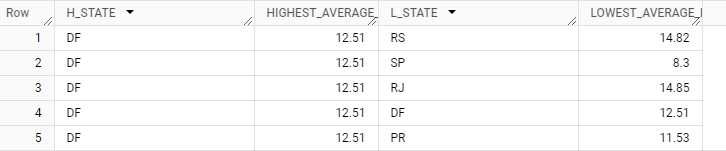
GROUP BY C.customer\_state ORDER BY C.customer\_state ASC

)

SELECT H\_STATE, HIGHEST\_AVERAGE\_DELIVERY, L\_STATE, LOWEST\_AVERAGE\_DELIVERY

FROM HIGH, LOW LIMIT 5;

# OUTPUT:-



## 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.**

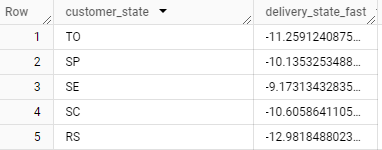
SELECT

C.customer\_state,AVG(DATE\_DIFF(order\_delivered\_customer\_date,order\_e stimated\_delivery\_date,DAY)) AS `delivery\_state\_fast`

FROM `Target\_sql.customers` AS C INNER JOIN `Target\_sql.orders` AS O ON C.customer\_id = O.customer\_id

GROUP BY C.customer\_state ORDER BY C.customer\_state DESC LIMIT 5;

# OUTPUT:-



## 6.Analysis based on the payments:

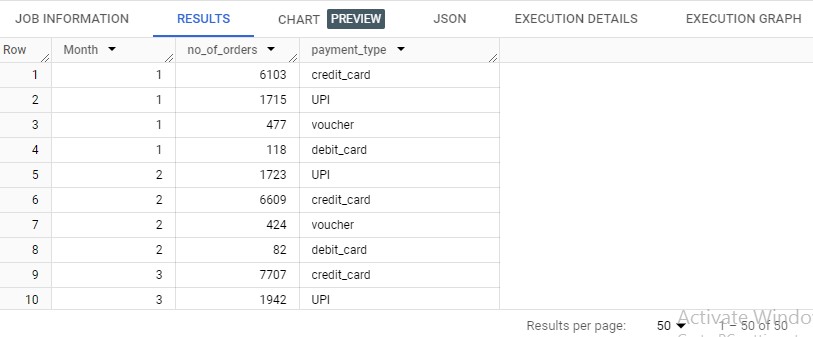
1. **Find the month on month no. of orders placed using different payment types.**

SELECT EXTRACT (MONTH FROM O.order\_purchase\_timestamp) AS `Month` , COUNT(O.order\_id) AS `no\_of\_orders`,P.payment\_type

FROM `Target\_sql.orders` AS O INNER JOIN `Target\_sql.payments` AS P On O.order\_id= P.order\_id

GROUP BY Month, P.payment\_type Order BY Month;

# OUTPUT:-



**INSIGHTS:-**

* + In every month credit card payments orders are high when compared other modes of payments.
  + In every month debit card payments orders are low when compared other modes of payments.

**RECOMMENDATIONS:-**

* + If we give credit card offers for some particular amount then there is a chance of getting more orders.
  + And for debit cards if we give some cash back while ordering then there is a chance of getting more orders.

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

SELECT COUNT(O.order\_id) AS `no\_of\_orders`,P.payment\_sequential FROM `Target\_sql.orders` AS O RIGHT JOIN `Target\_sql.payments` AS P On O.order\_id= P.order\_id

GROUP BY P.payment\_sequential;

# OUTPUT:-

