Name: **ABHISHEK SHETE**

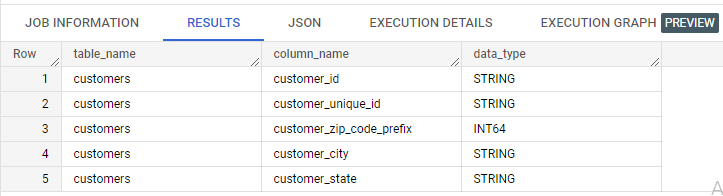
**Target test**

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

SELECT table\_name, column\_name, data\_type

FROM Target\_test.INFORMATION\_SCHEMA.COLUMNS

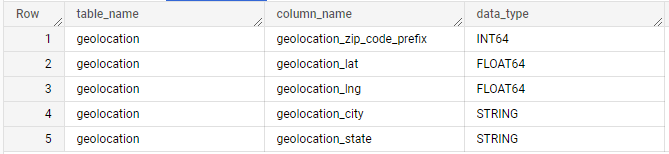
where table\_name = 'customers'



SELECT table\_name, column\_name, data\_type

FROM Target\_test.INFORMATION\_SCHEMA.COLUMNS

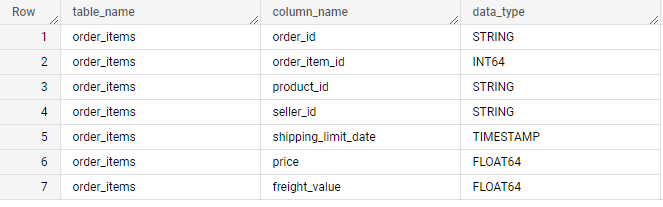
where table\_name = 'geolocation'



SELECT table\_name, column\_name, data\_type

FROM Target\_test.INFORMATION\_SCHEMA.COLUMNS

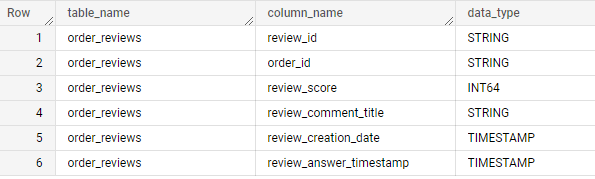
where table\_name = 'order\_items'



SELECT table\_name, column\_name, data\_type

FROM Target\_test.INFORMATION\_SCHEMA.COLUMNS

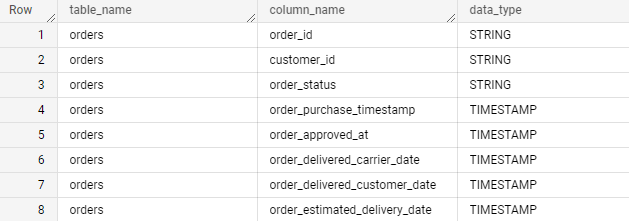
where table\_name = 'order\_reviews'



SELECT table\_name, column\_name, data\_type

FROM Target\_test.INFORMATION\_SCHEMA.COLUMNS

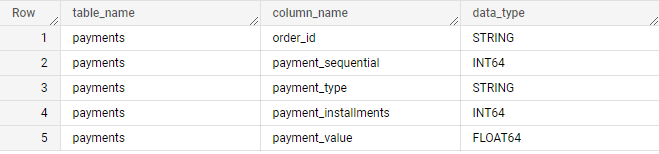
where table\_name = 'orders'



SELECT table\_name, column\_name, data\_type

FROM Target\_test.INFORMATION\_SCHEMA.COLUMNS

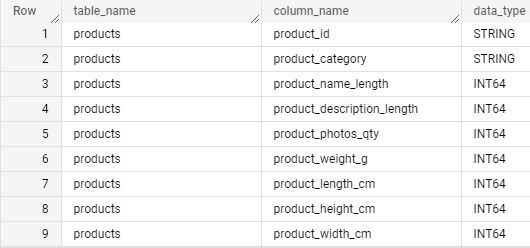
where table\_name = 'payments'



SELECT table\_name, column\_name, data\_type

FROM Target\_test.INFORMATION\_SCHEMA.COLUMNS

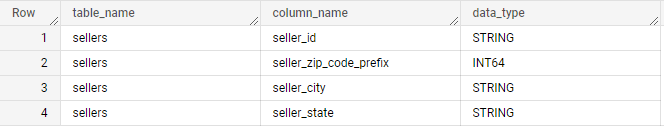
where table\_name = 'products'



SELECT table\_name, column\_name, data\_type

FROM Target\_test.INFORMATION\_SCHEMA.COLUMNS

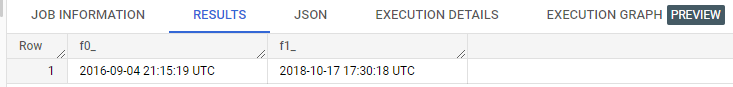
where table\_name = 'sellers'



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

SELECT MIN(order\_purchase\_timestamp), MAX(order\_purchase\_timestamp)

FROM Target\_test.orders



* 1. **Cities and States of customers ordered during the given period**

select distinct customer\_city as City, customer\_state as State

from `Target\_test.customers`

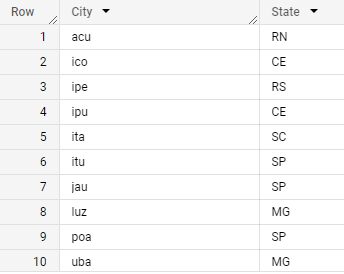
where customer\_id IN

(

select customer\_id

 from `Target\_test.orders`

)



2. In-depth Exploration:

2.1 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 EXTRACT(YEAR FROM DATETIME(order\_purchase\_timestamp)) as Year,

EXTRACT(MONTH FROM DATETIME(order\_purchase\_timestamp)) as Month,

COUNT(order\_id) AS Orders\_placed

from `Target\_test.orders`

GROUP BY 1,2

ORDER BY 1,2



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

select

case

 when Hour <= 6 then 'DAWN'

 when 7 <= Hour AND Hour <= 12 then 'MORNING'

 when 13 <= Hour AND Hour <= 18 then 'AFTERNOON'

 when 19 <= Hour AND Hour <= 23 then 'NIGHT'

 END as Time\_of\_day, SUM(Orders) as Orders\_placed

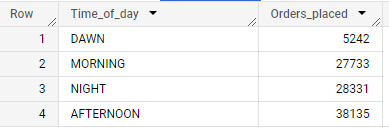
FROM (select count(\*) as Orders, EXTRACT(Hour from datetime(order\_purchase\_timestamp)) as Hour

from `Target\_test.orders`

GROUP BY 2)

GROUP BY 1

ORDER BY 2



The above table shows the maximum orders placed in “AFTERNOON”

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

3.1 Get month on month orders by states.

SELECT C.customer\_state AS state,EXTRACT(month FROM

O.order\_purchase\_timestamp) AS month, COUNT(O.order\_id) AS orders\_count

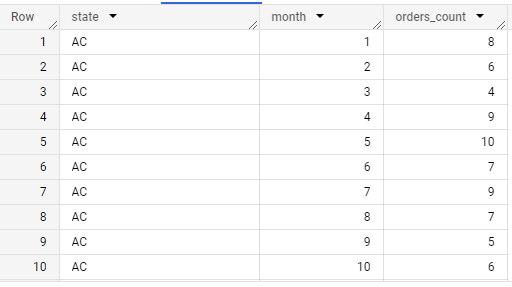
FROM `Target\_test.orders` O

JOIN `Target\_test.customers` C

ON O.customer\_id = C.customer\_id

group by 1,2

ORDER BY STATE, MONTH

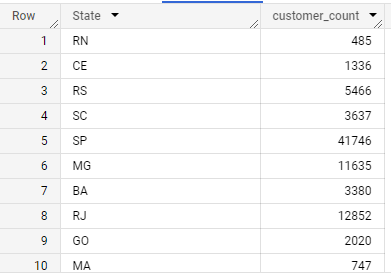


3.2 Distribution of customers across the states in Brazil

select customer\_state as State, count(customer\_id) as customer\_count

from Target\_test.customers

GROUP BY 1



4. Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.

4.1 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

WITH tvalue AS (

SELECT ROUND(SUM(payment\_value)) AS VAL, EXTRACT(YEAR FROM

o.order\_purchase\_timestamp) AS YEAR

FROM `Target\_test.payments` p

JOIN `Target\_test.orders` O

ON P.order\_id = O.order\_id

WHERE EXTRACT(MONTH FROM O.order\_purchase\_timestamp) IN (1,2,3,4,5,6,7,8)

 AND

 EXTRACT(YEAR FROM O.order\_purchase\_timestamp) IN (2017,2018)

GROUP BY 2

ORDER BY 2

)

SELECT ROUND((((SUM(t2.val) - SUM(t1.val)) / SUM(t1.val)) \* 100), 2) AS

increase\_percentage

FROM tvalue t1, tvalue t2

WHERE t1.Year = 2017 AND t2.Year = 2018



4.2 Mean & Sum of price and freight value by customer state

SELECT c.customer\_state AS State,

 ROUND(SUM(oi.price),2) AS sum\_price, ROUND(AVG(oi.price),2) AS mean\_price,

 ROUND(SUM(oi.freight\_value),2) AS sum\_freight,

ROUND(AVG(oi.freight\_value),2) AS mean\_freight

FROM `Target\_test.order\_items` oi

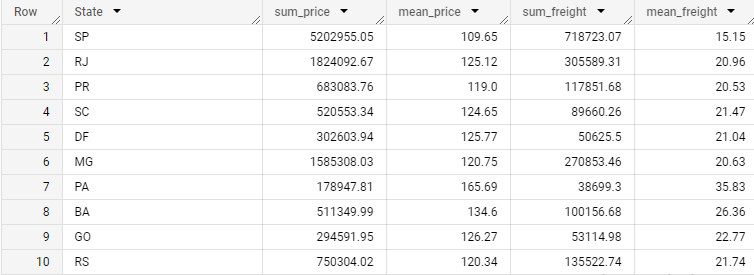
JOIN `Target\_test.orders` o

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

JOIN `Target\_test.customers` c

ON o.customer\_id = c.customer\_id

GROUP BY State



5. Analysis on sales, freight and delivery time

5.1 Calculate days between purchasing, delivering and estimated delivery

SELECT order\_id,

 DATE\_DIFF( order\_delivered\_customer\_date, order\_purchase\_timestamp,

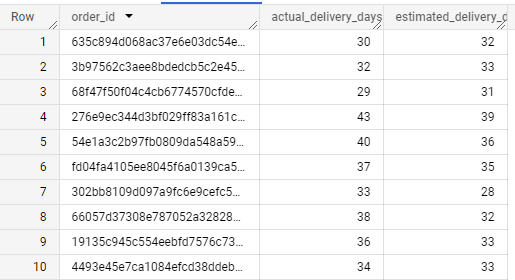
day) AS actual\_delivery\_days,

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

AS estimated\_delivery\_days

FROM `Target\_test.orders`

WHERE order\_status = 'delivered'



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

\* Time\_to\_delivery = order\_delivered\_customer\_date-order\_purchase\_timestamp

\* diff\_estimated\_delivery = order\_estimated\_delivery\_date order\_delivered\_customer\_date

SELECT order\_id,

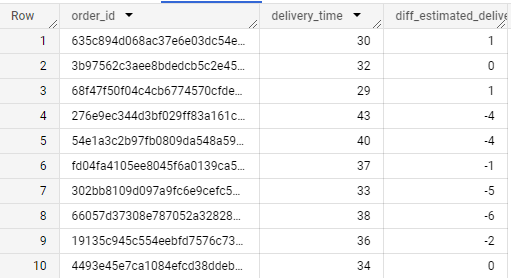
DATE\_DIFF(order\_delivered\_customer\_date, order\_purchase\_timestamp, DAY) AS delivery\_time,

DATE\_DIFF(order\_estimated\_delivery\_date, order\_delivered\_customer\_date,

DAY) AS diff\_estimated\_delivery

FROM `Target\_test.orders`

WHERE order\_status = 'delivered'



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

SELECT c.customer\_state AS STATE,

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

ROUND(AVG(DATE\_DIFF(order\_delivered\_customer\_date,

order\_purchase\_timestamp, DAY))) AS mean\_time\_to\_delivery,

ROUND(AVG(DATE\_DIFF(order\_estimated\_delivery\_date,

order\_delivered\_customer\_date, DAY))) AS mean\_diff\_estimated\_delivery

FROM `Target\_test.customers` c

JOIN `Target\_test.orders` o

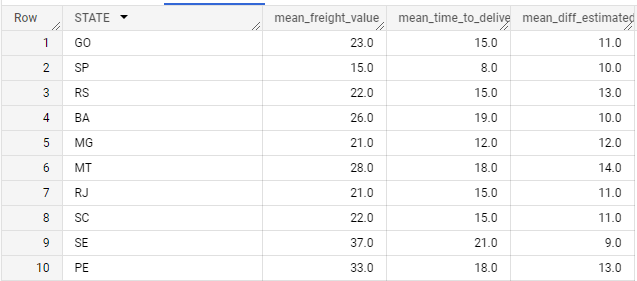
ON c.customer\_id = o.customer\_id

JOIN `Target\_test.order\_items` oi

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

WHERE o.order\_status = 'delivered'

GROUP BY 1



5.4 Sort the data to get the following:

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

SELECT c.customer\_state AS STATE,

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

 ROUND(AVG(DATE\_DIFF(order\_delivered\_customer\_date,

order\_purchase\_timestamp, DAY))) AS mean\_time\_to\_delivery,

 ROUND(AVG(DATE\_DIFF(order\_estimated\_delivery\_date,

order\_delivered\_customer\_date, DAY))) AS mean\_diff\_estimated\_delivery

FROM `Target\_test.customers` c

JOIN `Target\_test.orders` o

ON c.customer\_id = o.customer\_id

JOIN `Target\_test.order\_items` oi

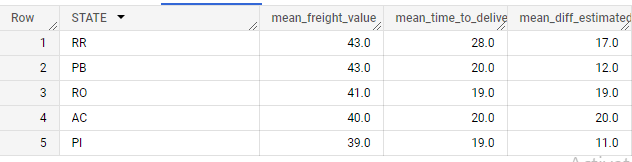
ON o.order\_id = oi.order\_id

WHERE o.order\_status = 'delivered'

GROUP BY 1

ORDER BY 2 DESC

LIMIT 5



SELECT c.customer\_state AS STATE,

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

 ROUND(AVG(DATE\_DIFF(order\_delivered\_customer\_date,

order\_purchase\_timestamp, DAY))) AS mean\_time\_to\_delivery,

 ROUND(AVG(DATE\_DIFF(order\_estimated\_delivery\_date,

order\_delivered\_customer\_date, DAY))) AS mean\_diff\_estimated\_delivery

FROM `Target\_test.customers` c

JOIN `Target\_test.orders` o

ON c.customer\_id = o.customer\_id

JOIN `Target\_test.order\_items` oi

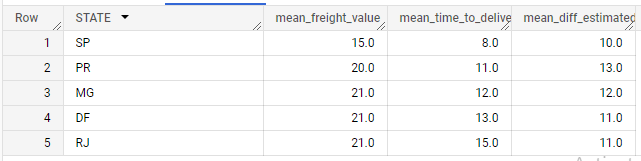
ON o.order\_id = oi.order\_id

WHERE o.order\_status = 'delivered'

GROUP BY 1

ORDER BY 2

LIMIT 5



5.6 Top 5 states with highest/lowest average time to delivery.

SELECT c.customer\_state AS STATE,

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

 ROUND(AVG(DATE\_DIFF(order\_delivered\_customer\_date,

order\_purchase\_timestamp, DAY))) AS mean\_time\_to\_delivery,

 ROUND(AVG(DATE\_DIFF(order\_estimated\_delivery\_date,

order\_delivered\_customer\_date, DAY))) AS mean\_diff\_estimated\_delivery

FROM `Target\_test.customers` c

JOIN `Target\_test.orders` o

ON c.customer\_id = o.customer\_id

JOIN `Target\_test.order\_items` oi

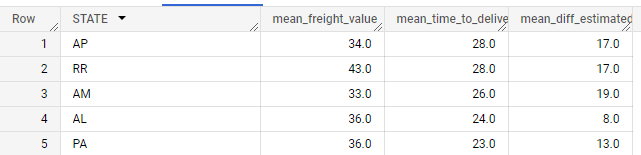
ON o.order\_id = oi.order\_id

WHERE o.order\_status = 'delivered'

GROUP BY 1

ORDER BY 3 DESC

LIMIT 5



SELECT c.customer\_state AS STATE,

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

 ROUND(AVG(DATE\_DIFF(order\_delivered\_customer\_date,

order\_purchase\_timestamp, DAY))) AS mean\_time\_to\_delivery,

 ROUND(AVG(DATE\_DIFF(order\_estimated\_delivery\_date,

order\_delivered\_customer\_date, DAY))) AS mean\_diff\_estimated\_delivery

FROM `Target\_test.customers` c

JOIN `Target\_test.orders` o

ON c.customer\_id = o.customer\_id

JOIN `Target\_test.order\_items` oi

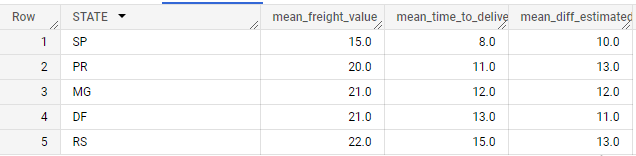
ON o.order\_id = oi.order\_id

WHERE o.order\_status = 'delivered'

GROUP BY 1

ORDER BY 3

LIMIT 5



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

SELECT c.customer\_state AS STATE,

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

 ROUND(AVG(DATE\_DIFF(order\_delivered\_customer\_date,

order\_purchase\_timestamp, DAY))) AS mean\_time\_to\_delivery,

 ROUND(AVG(DATE\_DIFF(order\_estimated\_delivery\_date,

order\_delivered\_customer\_date, DAY))) AS mean\_diff\_estimated\_delivery

FROM `Target\_test.customers` c

JOIN `Target\_test.orders` o

ON c.customer\_id = o.customer\_id

JOIN `Target\_test.order\_items` oi

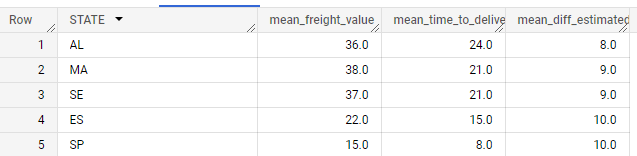
ON o.order\_id = oi.order\_id

WHERE o.order\_status = 'delivered'

GROUP BY 1

ORDER BY 4

LIMIT 5



SELECT c.customer\_state AS STATE,

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

 ROUND(AVG(DATE\_DIFF(order\_delivered\_customer\_date,

order\_purchase\_timestamp, DAY))) AS mean\_time\_to\_delivery,

 ROUND(AVG(DATE\_DIFF(order\_estimated\_delivery\_date,

order\_delivered\_customer\_date, DAY))) AS mean\_diff\_estimated\_delivery

FROM `Target\_test.customers` c

JOIN `Target\_test.orders` o

ON c.customer\_id = o.customer\_id

JOIN `Target\_test.order\_items` oi

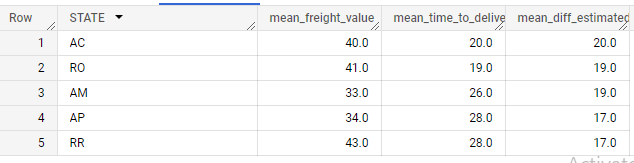
ON o.order\_id = oi.order\_id

WHERE o.order\_status = 'delivered'

GROUP BY 1

ORDER BY 4 DESC

LIMIT 5



6. Payment type analysis:

6.1 Month over Month count of orders for different payment types

SELECT EXTRACT(MONTH FROM o.order\_purchase\_timestamp) AS MONTH,

P.payment\_type, COUNT(O.order\_id) AS Order\_count

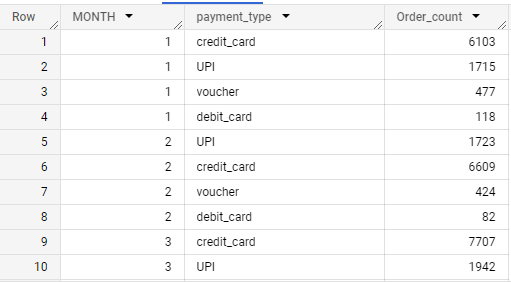
FROM `Target\_test.orders` O

JOIN `Target\_test.payments` P

ON O.order\_id = P.order\_id

GROUP BY 1,2

ORDER BY 1



6.2 Count of orders based on the no. of payment installments.

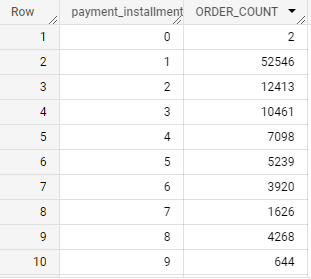
SELECT P.payment\_installments, COUNT(O.order\_id) AS ORDER\_COUNT

FROM `Target\_test.orders` O

JOIN `Target\_test.payments` P

ON O.order\_id = P.order\_id

GROUP BY 1 ORDER BY 1



7. Action Insights:

7.1 All and all the data was provided from the date between (2016-09-04 21:15:19 UTC and 2018-10-17 17:30:18 UTC).

7.2 Most of the time orders placed in afternoon.

7.3 136.98 % increase in cost of orders from year 2017 to 2018 as per the data.

7.4 Credit card is most preferred payment option, compared with the credit card which is less preferred.

8. Recommendations:

8.1 the customers who buy the products(orders) with single or maximum three installments they get discounts like cash backs, redeem coupons, offers.

8.2 As shown the majority of the orders are done by credit cards provides special offers like cash backs.

8.3 Giving discounts on orders when there is fall in sales.

8.4 Most of the orders are placed in afternoon time so that flash sales, bonus offers, short term discounts should be provided during the same time.