**Table of Contents**

[1. Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset 2](#__RefHeading___Toc57_1618691585)

[1. Data type of columns in a table 2](#__RefHeading___Toc59_1618691585)

[2. Time period for which the data is given 2](#__RefHeading___Toc61_1618691585)

[3. Cities and States of customers ordered during the given period 2](#__RefHeading___Toc63_1618691585)

[2. In-depth Exploration: 3](#__RefHeading___Toc65_1618691585)

[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? 3](#__RefHeading___Toc67_1618691585)

[2. What time do Brazilian customers tend to buy (Dawn, Morning, Afternoon or Night)? 4](#__RefHeading___Toc69_1618691585)

[3. Evolution of E-commerce orders in the Brazil region: 5](#__RefHeading___Toc71_1618691585)

[1. Get month on month orders by states 5](#__RefHeading___Toc73_1618691585)

[2. Distribution of customers across the states in Brazil 5](#__RefHeading___Toc75_1618691585)

[4.Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others. 6](#__RefHeading___Toc77_1618691585)

[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 6](#__RefHeading___Toc79_1618691585)

[2. Mean & Sum of price and freight value by customer state 6](#__RefHeading___Toc81_1618691585)

[5. Analysis on sales, freight and delivery time 7](#__RefHeading___Toc83_1618691585)

[1. Calculate days between purchasing, delivering and estimated delivery 7](#__RefHeading___Toc85_1618691585)

[2. Find time\_to\_delivery & diff\_estimated\_delivery. Formula for the same given below: 7](#__RefHeading___Toc87_1618691585)

[a) time\_to\_delivery = order\_purchase\_timestamp-order\_delivered\_customer\_date 7](#__RefHeading___Toc89_1618691585)

[b) diff\_estimated\_delivery = order\_estimated\_delivery\_date-order\_delivered\_customer\_date 7](#__RefHeading___Toc91_1618691585)

[3. Group data by state, take mean of freight\_value, time\_to\_delivery, diff\_estimated\_delivery 8](#__RefHeading___Toc93_1618691585)

[4. Sort the data to get the following: 8](#__RefHeading___Toc95_1618691585)

[a) Top 5 states with highest/lowest average freight value - sort in desc/asc limit 5 8](#__RefHeading___Toc97_1618691585)

[b) Top 5 states with highest/lowest average time to delivery 9](#__RefHeading___Toc99_1618691585)

[c) Top 5 states where delivery is really fast/ not so fast compared to estimated date 9](#__RefHeading___Toc101_1618691585)

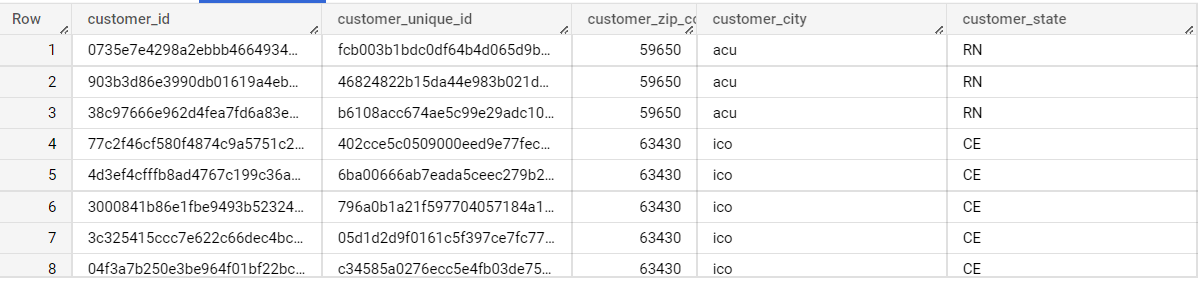
[6. Payment type analysis: 10](#__RefHeading___Toc103_1618691585)

[1. Month over Month count of orders for different payment types 10](#__RefHeading___Toc105_1618691585)

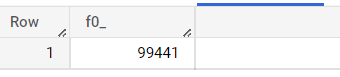
[2. Count of orders based on the no. of payment installments 10](#__RefHeading___Toc107_1618691585)

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

select \* from `target-project-373409.Target.customers` limit 10

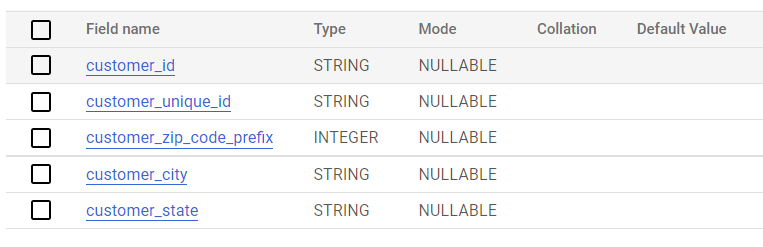


select count(\*) from `target-project-373409.Target.customers`



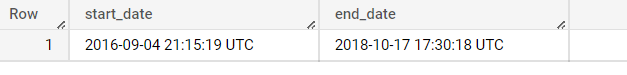
## 1. Data type of columns in a table

Describe `target-project-373409.Target.customers`



## 2. Time period for which the data is given

SELECT min(order\_purchase\_timestamp) as start\_date, max(order\_purchase\_timestamp) end\_date FROM `target-project-373409.Target.orders`



## 3. Cities and States of customers ordered during the given period

SELECT

c.customer\_city,c.customer\_state

FROM

  `target-project-373409.Target.orders`o

inner join

`target-project-373409.Target.customers`c

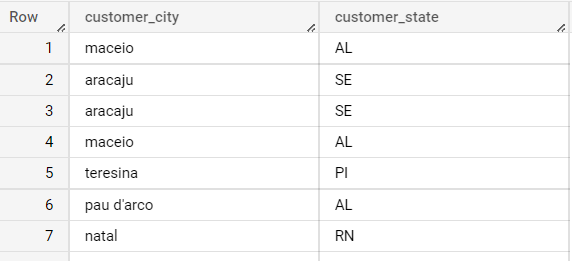
on o.customer\_id = c.customer\_id

where o.order\_purchase\_timestamp between '2016-09-04 21:15:19'

and

'2018-10-17 17:30:18'

and UPPER(o.order\_status) = ‘delivered’;



# 2. In-depth Exploration:

## 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(\*) order\_count,

  EXTRACT(year

  FROM order\_delivered\_customer\_date) year ,

  EXTRACT(month

  FROM

    order\_delivered\_customer\_date) month

FROM

  `target-project-373409.Target.orders`

WHERE

  order\_status = 'delivered'

GROUP BY

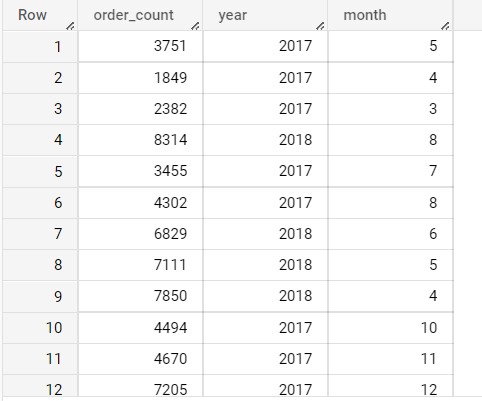
  EXTRACT(year

  FROM

    order\_delivered\_customer\_date) , EXTRACT(month

  FROM

    order\_delivered\_customer\_date)



SELECT

  COUNT(\*) order\_count,

  EXTRACT(month

  FROM

    order\_delivered\_customer\_date) as month

FROM

  `target-project-373409.Target.orders`

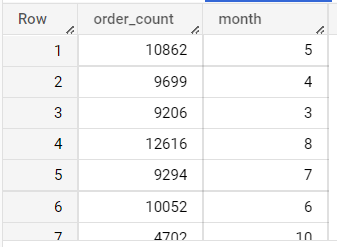
WHERE

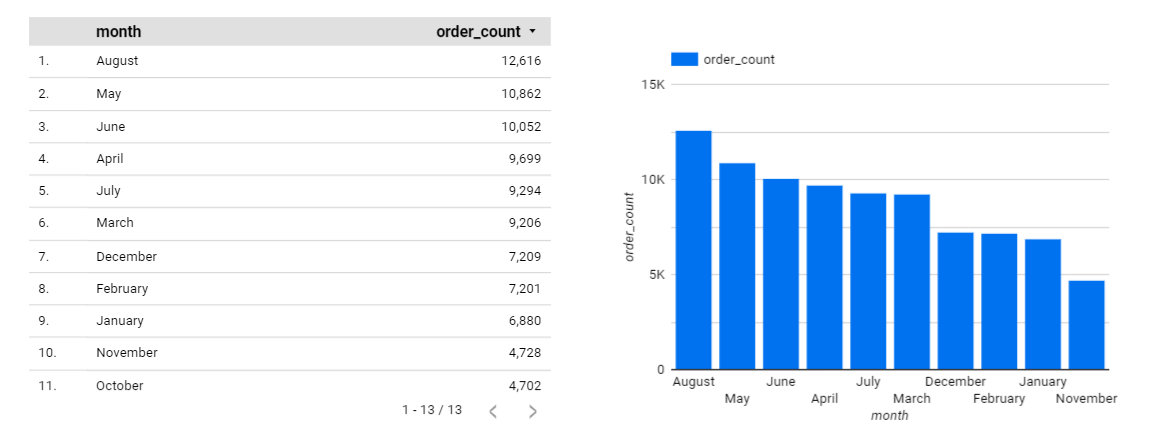
  order\_status = 'delivered'

GROUP BY EXTRACT(month

  FROM

    order\_delivered\_customer\_date)





August month has highest orders

SELECT

  COUNT(\*) order\_count,

  EXTRACT(year

  FROM

    order\_delivered\_customer\_date) as year

FROM

  `target-project-373409.Target.orders`

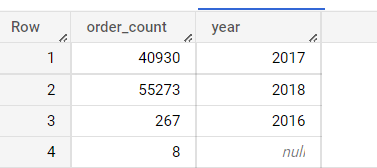
WHERE

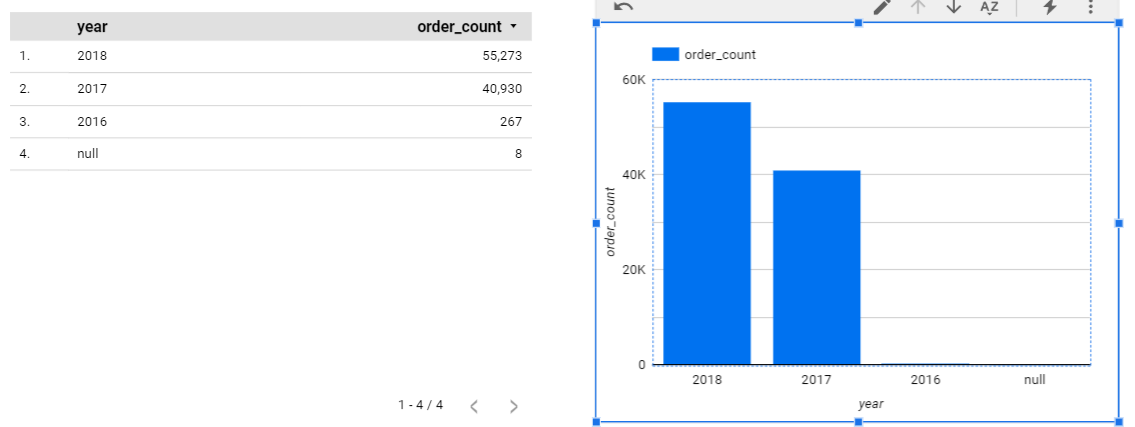
  order\_status = 'delivered'

GROUP BY EXTRACT(year

  FROM

    order\_delivered\_customer\_date)





Order Is gradually increasing over the year

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

with data as (

SELECT

  COUNT(\*) order\_count,

extract(hour from order\_purchase\_timestamp ) as time\_of\_purchase

FROM

  `target-project-373409.Target.orders`

  group by extract(hour from order\_purchase\_timestamp )),

  period as

  (select data.\*,

  case when time\_of\_purchase between 0 and 5 then 'dawn'

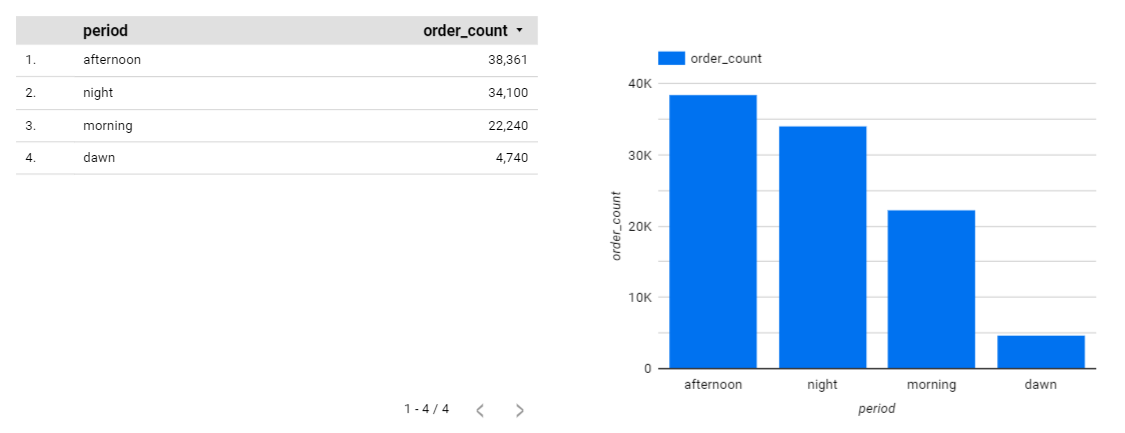
   when time\_of\_purchase between 6 and 11 then 'morning'

  when time\_of\_purchase between 12 and 17 then 'afternoon'

 when time\_of\_purchase between 18 and 23 then 'night'

   end as period from data)

   select \*  from period  ;



with data as (

SELECT

  COUNT(\*) order\_count,

extract(hour from order\_purchase\_timestamp ) as time\_of\_purchase

FROM

  `target-project-373409.Target.orders`

  group by extract(hour from order\_purchase\_timestamp )),

  period as

  (select data.\*,

sum( case when time\_of\_purchase between 0 and 5 then  order else 0 end ) as ‘dawn’

sum( case when time\_of\_purchase between 0 and 5 then  orders else 0 end ) as 'morning'

sum( case when time\_of\_purchase between 0 and 5 then  order else 0 end ) as ‘afternoon'

sum( case when time\_of\_purchase between 0 and 5 then  order else 0 end ) as 'night'

   select \*  from period  ;

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

## Get month on month orders by states

SELECT c.customer\_state, extract(year from order\_purchase\_timestamp) year ,extract(month from order\_purchase\_timestamp) month

,count(o.order\_id)count\_order

FROM `target-project-373409.Target.customers`c

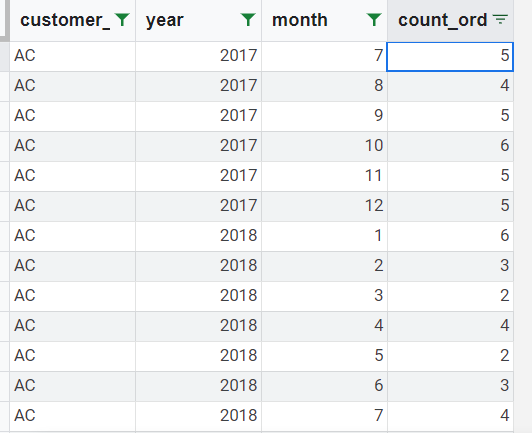
join `target-project-373409.Target.orders` o

on c.customer\_id = o.customer\_id

group by c.customer\_state,extract(year from order\_purchase\_timestamp),

extract(month from order\_purchase\_timestamp)

order by c.customer\_state,extract(year from order\_purchase\_timestamp)  ,extract(month from order\_purchase\_timestamp)



## Distribution of customers across the states in Brazil

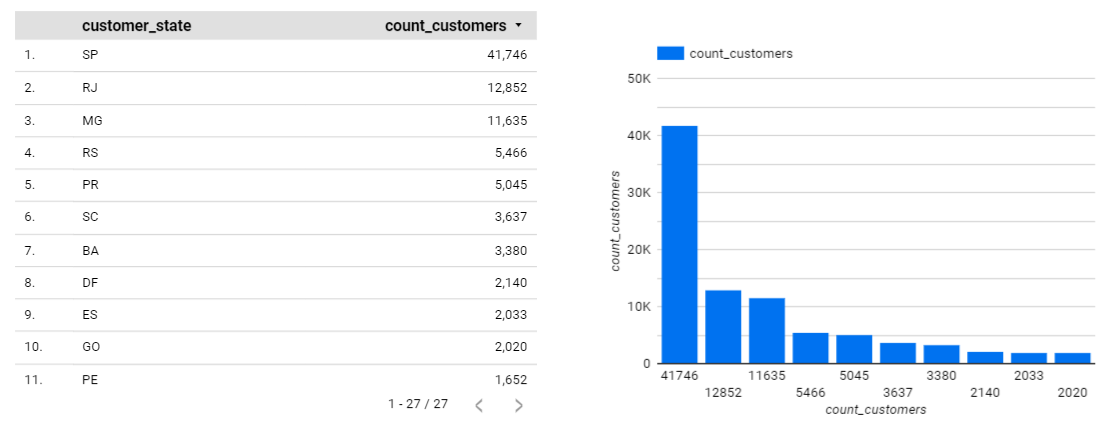
SELECT customer\_state

,count(customer\_id)count\_customers

FROM `target-project-373409.Target.customers`

group by customer\_state

order by count\_customers desc



# 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

SELECT

  EXTRACT(year FROM order\_purchase\_timestamp) year,

  EXTRACT(month FROM order\_purchase\_timestamp) month,

  SUM(payment\_value) total\_cost,

  CASE WHEN LAG(SUM(payment\_value)) OVER (PARTITION BY EXTRACT(year FROM order\_purchase\_timestamp) ORDER BY EXTRACT(month FROM order\_purchase\_timestamp)) = 0 THEN 0

    ELSE

  (SUM(payment\_value) - LAG(SUM(payment\_value)) OVER (PARTITION BY EXTRACT(year FROM order\_purchase\_timestamp) ORDER BY EXTRACT(month FROM order\_purchase\_timestamp))) / LAG(SUM(payment\_value)) OVER (PARTITION BY EXTRACT(year FROM order\_purchase\_timestamp) ORDER BY EXTRACT(month FROM order\_purchase\_timestamp)) \* 100 END percentage\_increase

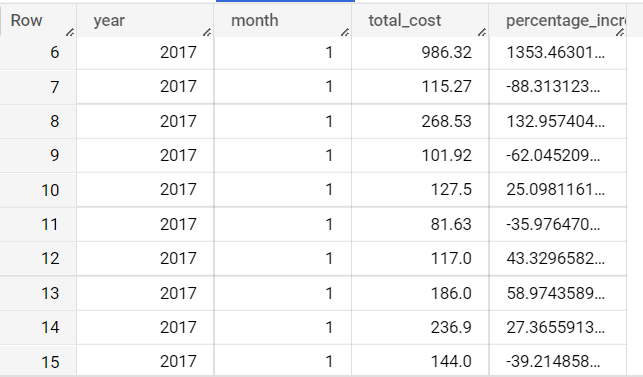
FROM `target-project-373409.Target.payments`p join `target-project-373409.Target.orders` o on p.order\_id = o.order\_id

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

AND EXTRACT(month FROM order\_purchase\_timestamp) BETWEEN 1 AND 8

GROUP BY EXTRACT(year FROM order\_purchase\_timestamp),EXTRACT(month FROM order\_purchase\_timestamp), order\_purchase\_timestamp

ORDER BY year ASC, month ASC;

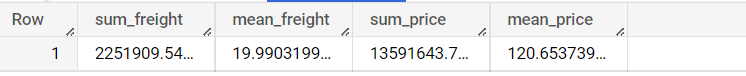


## 2. Mean & Sum of price and freight value by customer state

SELECT sum(freight\_value) sum\_freight,

avg(freight\_value) mean\_freight,

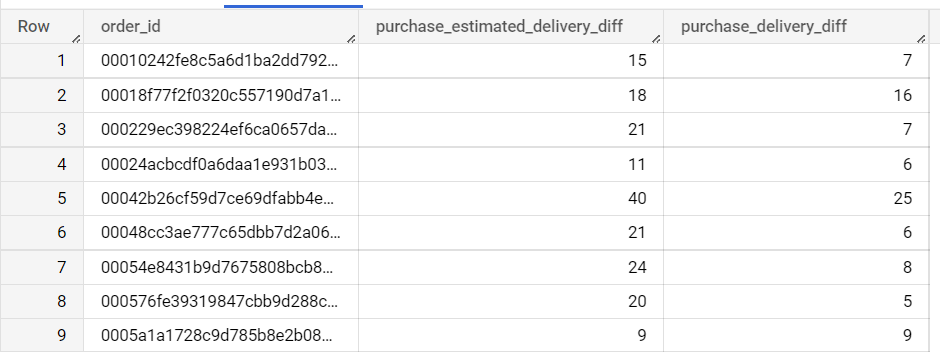
sum(price) sum\_price, avg(price) mean\_price FROM `target-project-373409.Target.order\_items` oi  join `target-project-373409.Target.orders` o on oi.order\_id = o.order\_id join `target-project-373409.Target.customers`c on c.customer\_id = o.customer\_id



# 5. Analysis on sales, freight and delivery time

## Calculate days between purchasing, delivering and estimated delivery

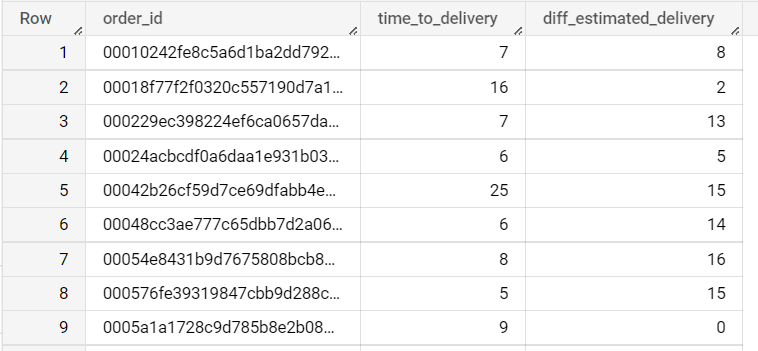
SELECT  order\_id, date\_diff(order\_estimated\_delivery\_date, order\_purchase\_timestamp, day) purchase\_estimated\_delivery\_diff, date\_diff(order\_delivered\_customer\_date, order\_purchase\_timestamp, day) purchase\_delivery\_diff FROM `target-project-373409.Target.order` order by order\_id



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

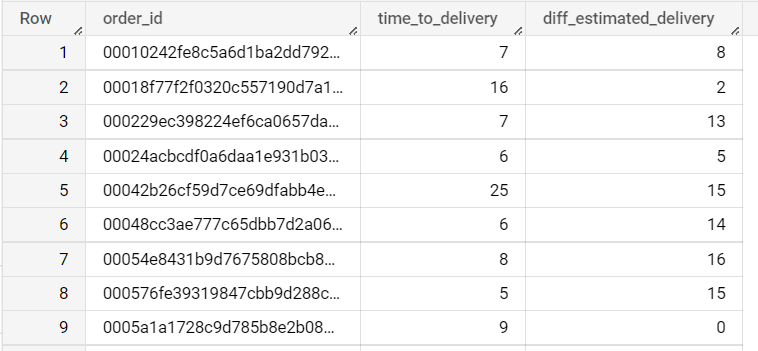
## time\_to\_delivery = order\_purchase\_timestamp-order\_delivered\_customer\_date

SELECT  order\_id, date\_diff(order\_delivered\_customer\_date, order\_purchase\_timestamp, day) time\_to\_delivery , date\_diff(order\_estimated\_delivery\_date, order\_delivered\_customer\_date, day) diff\_estimated\_delivery FROM `target-project-373409.Target.order` order by order\_id



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

SELECT  order\_id, date\_diff(order\_delivered\_customer\_date, order\_purchase\_timestamp, day) time\_to\_delivery , date\_diff(order\_estimated\_delivery\_date, order\_delivered\_customer\_date, day) diff\_estimated\_delivery FROM `target-project-373409.Target.order` order by order\_id

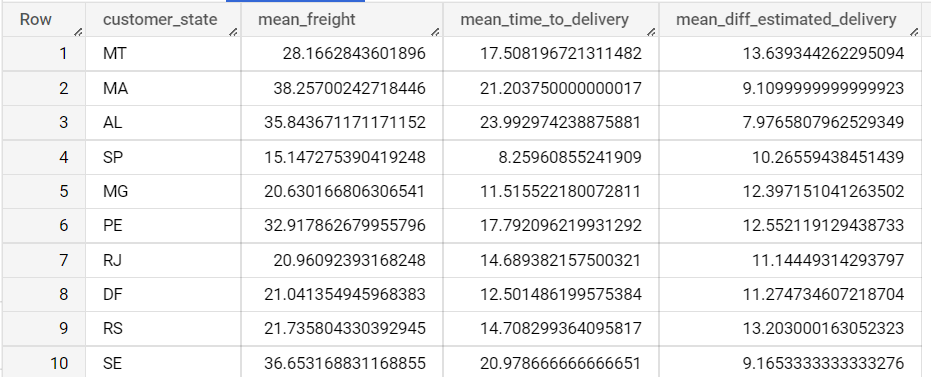


## Group data by state, take mean of freight\_value, time\_to\_delivery, diff\_estimated\_delivery

select

customer\_state, avg(freight\_value) mean\_freight,avg(date\_diff(order\_delivered\_customer\_date, order\_purchase\_timestamp, day)) mean\_time\_to\_delivery ,avg(date\_diff(order\_estimated\_delivery\_date, order\_delivered\_customer\_date, day)) mean\_diff\_estimated\_delivery

 FROM `target-project-373409.Target.order\_items` oi  join `target-project-373409.Target.orders` o on oi.order\_id = o.order\_id join `target-project-373409.Target.customers`c on c.customer\_id = o.customer\_id group by customer\_state



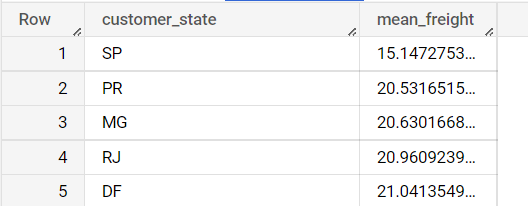
## Sort the data to get the following:

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

select

customer\_state, avg(freight\_value) mean\_freight

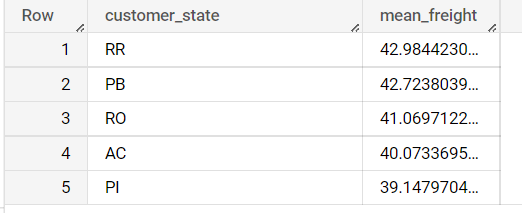
 FROM `target-project-373409.Target.order\_items` oi  join `target-project-373409.Target.orders` o on oi.order\_id = o.order\_id join `target-project-373409.Target.customers`c on c.customer\_id = o.customer\_id group by customer\_state order by mean\_freight limit 5



select

customer\_state, avg(freight\_value) mean\_freight

 FROM `target-project-373409.Target.order\_items` oi  join `target-project-373409.Target.orders` o on oi.order\_id = o.order\_id join `target-project-373409.Target.customers`c on c.customer\_id = o.customer\_id group by customer\_state order by mean\_freight desc limit 5

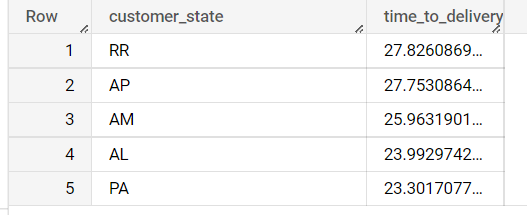


## Top 5 states with highest/lowest average time to delivery

select

customer\_state, avg(date\_diff(order\_delivered\_customer\_date, order\_purchase\_timestamp, day)) time\_to\_delivery

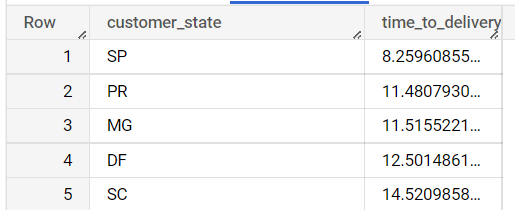
 FROM `target-project-373409.Target.order\_items` oi  join `target-project-373409.Target.orders` o on oi.order\_id = o.order\_id join `target-project-373409.Target.customers`c on c.customer\_id = o.customer\_id group by customer\_state order by time\_to\_delivery desc limit 5



select

customer\_state, avg(date\_diff(order\_delivered\_customer\_date, order\_purchase\_timestamp, day)) time\_to\_delivery

 FROM `target-project-373409.Target.order\_items` oi  join `target-project-373409.Target.orders` o on oi.order\_id = o.order\_id join `target-project-373409.Target.customers`c on c.customer\_id = o.customer\_id group by customer\_state order by time\_to\_delivery limit 5

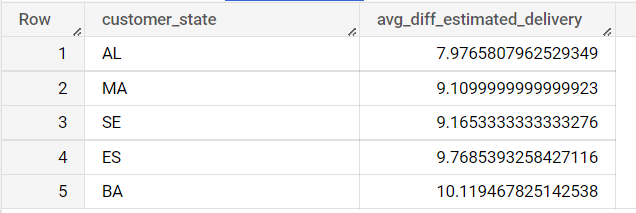


## Top 5 states where delivery is really fast/ not so fast compared to estimated date

select

customer\_state, avg(date\_diff(order\_estimated\_delivery\_date, order\_delivered\_customer\_date, day) )avg\_diff\_estimated\_delivery

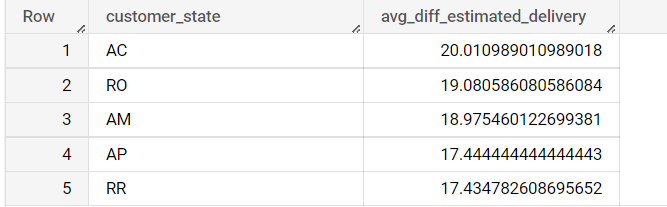
 FROM `target-project-373409.Target.order\_items` oi  join `target-project-373409.Target.orders` o on oi.order\_id = o.order\_id join `target-project-373409.Target.customers`c on c.customer\_id = o.customer\_id group by customer\_state order by avg\_diff\_estimated\_delivery  limit 5



select

customer\_state, avg(date\_diff(order\_estimated\_delivery\_date, order\_delivered\_customer\_date, day) )avg\_diff\_estimated\_delivery

 FROM `target-project-373409.Target.order\_items` oi  join `target-project-373409.Target.orders` o on oi.order\_id = o.order\_id join `target-project-373409.Target.customers`c on c.customer\_id = o.customer\_id group by customer\_state order by avg\_diff\_estimated\_delivery desc limit 5



# 6. Payment type analysis:

## Month over Month count of orders for different payment types

SELECT

  EXTRACT(month FROM o.order\_purchase\_timestamp) month,

  EXTRACT(year FROM o.order\_purchase\_timestamp) year,

  p.payment\_type payment\_type,

  COUNT(\*) order\_count

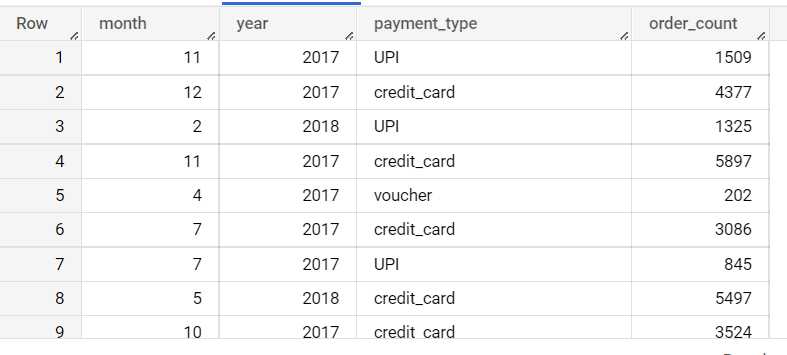
FROM `target-project-373409.Target.payments` as p  join `target-project-373409.Target.orders` as o on p.order\_id = o.order\_id

GROUP BY

  EXTRACT(month FROM o.order\_purchase\_timestamp),

  EXTRACT(year FROM o.order\_purchase\_timestamp),

  p.payment\_type



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

SELECT

 p.payment\_installments,

  COUNT(\*) order\_count

FROM `target-project-373409.Target.payments` as p  join `target-project-373409.Target.orders` as o on p.order\_id = o.order\_id

group by payment\_installments

order by payment\_installments

