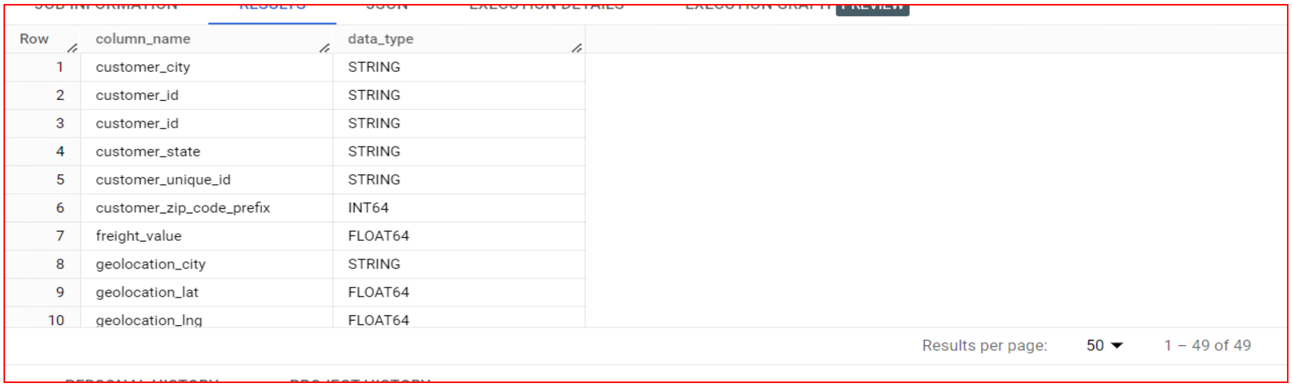
PROJECT

Q1(a) 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-382417.Target_sql_businesscase.INFORMATION_SCHEMA.COLUMNS
order by column_name
```

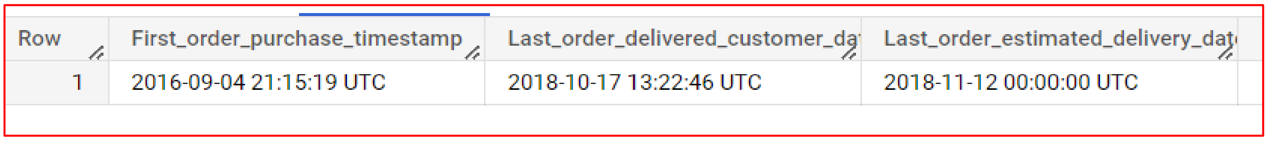


Row	column_name	data_type
1	customer_city	STRING
2	customer_id	STRING
3	customer_id	STRING
4	customer_state	STRING
5	customer_unique_id	STRING
6	customer_zip_code_prefix	INT64
7	freight_value	FLOAT64
8	geolocation_city	STRING
9	geolocation_lat	FLOAT64
10	geolocation_lng	FLOAT64

Results per page: 50 ▾ 1 – 49 of 49

2) Time period for which the data is given

```
select min(order_purchase_timestamp) as First_order_purchase_timestamp,
max(order_delivered_customer_date) as Last_order_delivered_customer_date ,
max(order_estimated_delivery_date) as Last_order_estimated_delivery_date
from `Target_sql_businesscase.orders`
```

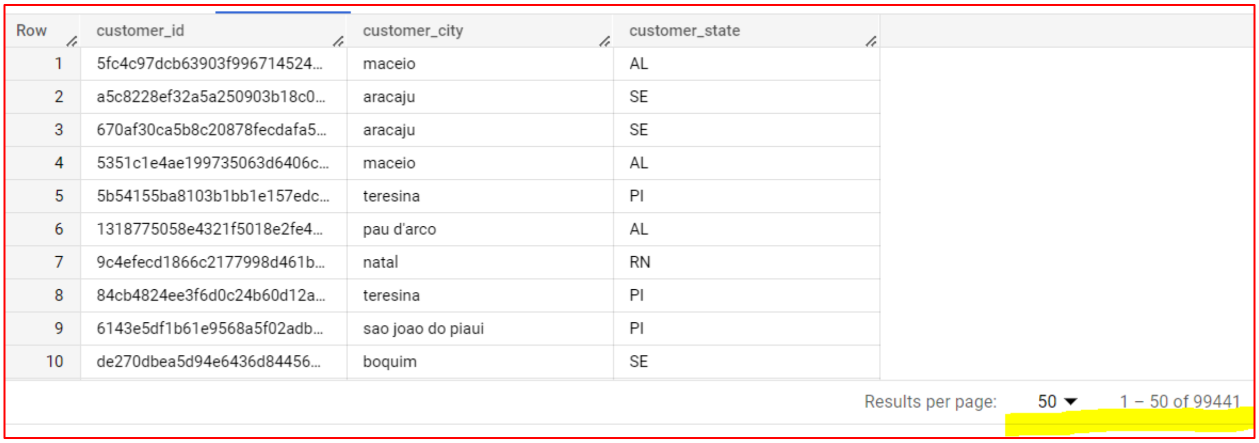


Row	First_order_purchase_timestamp	Last_order_delivered_customer_da	Last_order_estimated_delivery_dat
1	2016-09-04 21:15:19 UTC	2018-10-17 13:22:46 UTC	2018-11-12 00:00:00 UTC

(3) Cities and States of customers ordered during the given period

(i) Details of customers who ordered:

```
select c.customer_id,c.customer_city,c.customer_state
from `target-382417.Target_sql_businesscase.customers` as c
join `target-382417.Target_sql_businesscase.orders` as o
on c.customer_id=o.customer_id
```

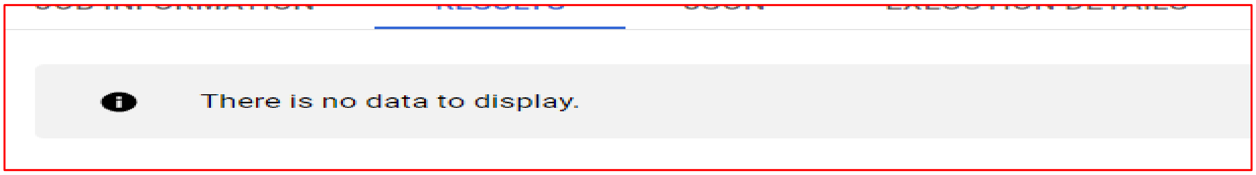


Row	customer_id	customer_city	customer_state
1	5fc4c97dcb63903f996714524...	maceio	AL
2	a5c8228ef32a5a250903b18c0...	aracaju	SE
3	670af30ca5b8c20878fecdafa5...	aracaju	SE
4	5351c1e4ae199735063d6406c...	maceio	AL
5	5b54155ba8103b1bb1e157edc...	teresina	PI
6	1318775058e4321f5018e2fe4...	pau d'arco	AL
7	9c4efecd1866c2177998d461b...	natal	RN
8	84cb4824ee3f6d0c24b60d12a...	teresina	PI
9	6143e5df1b61e9568a5f02adb...	sao joao do piaui	PI
10	de270dbea5d94e6436d84456...	boquim	SE

Results per page: 50 ▾ 1 – 50 of 99441

ii) Details of customers who have not ordered:

```
select c.customer_city,c.customer_state,o.order_id
from `target-382417.Target_sql_businesscase.customers` as c
left join `target-382417.Target_sql_businesscase.orders` as o
on c.customer_id=o.customer_id
where o.order_id is null
```



There is no data to display.

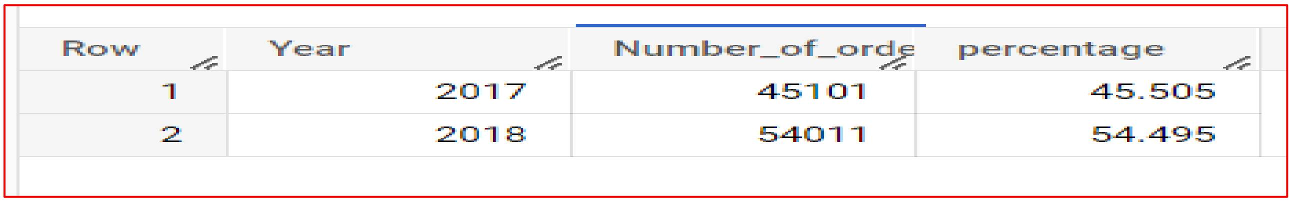
Conclusions:

```
    1)  Data is given for the time period :04-09-2016 to 17-10-2018

    2)  Every customer has made a purchase so we can find customer_city,customer_state
        from customer table only.

     select customer_city,customer_state
     from `target-382417.Target_sql_businesscase.customers` as c
```

Q(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?

Yearly growth:

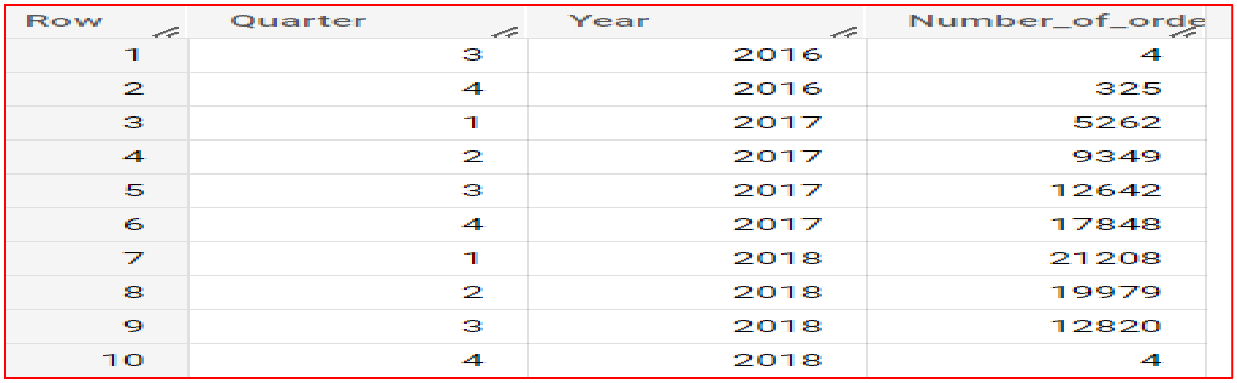
```
 with table as( select
extract(YEAR FROM order_purchase_timestamp) AS Year,
count(distinct order_id) as Number_of_orders,
sum(count(distinct order_id)) over () as total
from `target-382417.Target_sql_businesscase.orders`
where extract(YEAR FROM order_purchase_timestamp) !=2016
group by Year
order by Year)
select Year, Number_of_orders, round((Number_of_orders/total)*100,3) as percentage
from table
```



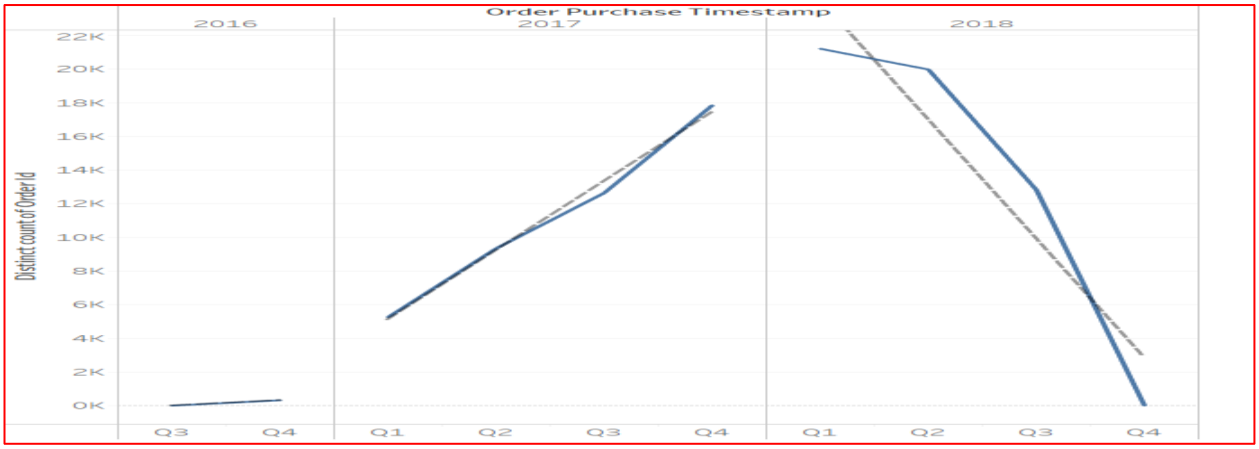
Row	Year	Number_of_orde	percentage
1	2017	45101	45.505
2	2018	54011	54.495

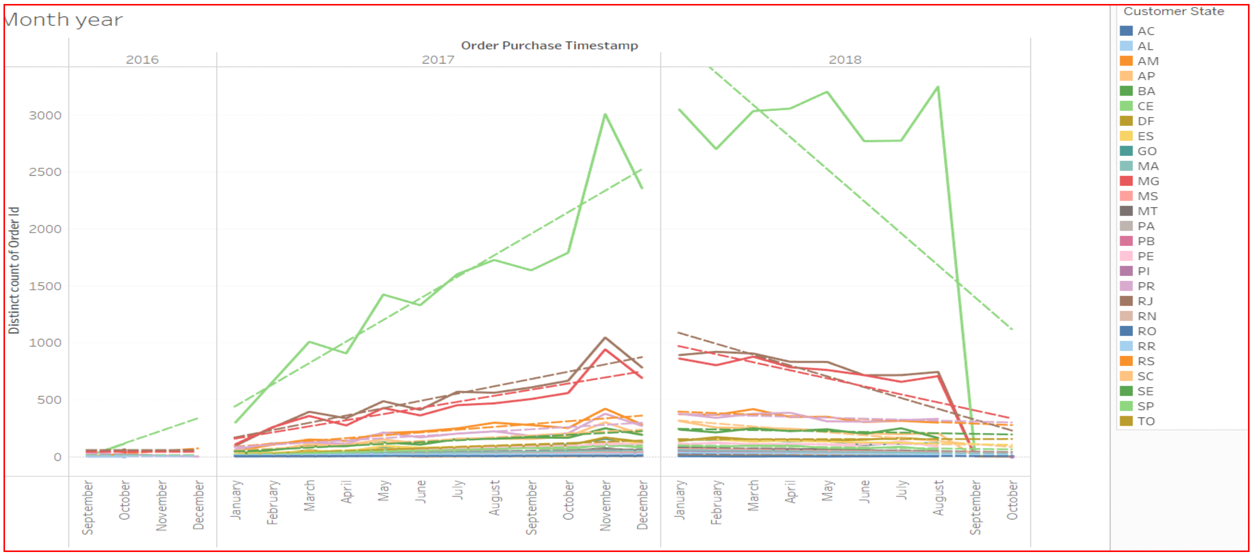
Quarterly growth:

```
 select extract(Quarter FROM order_purchase_timestamp) AS Quarter,
extract(YEAR FROM order_purchase_timestamp) AS Year,
count(distinct order_id) as Number_of_orders,
from `target-382417.Target_sql_businesscase.orders`
group by Quarter,Year
order by Year,Quarter
```



Row	Quarter	Year	Number_of_orde
1	3	2016	4
2	4	2016	325
3	1	2017	5262
4	2	2017	9349
5	3	2017	12642
6	4	2017	17848
7	1	2018	21208
8	2	2018	19979
9	3	2018	12820
10	4	2018	4



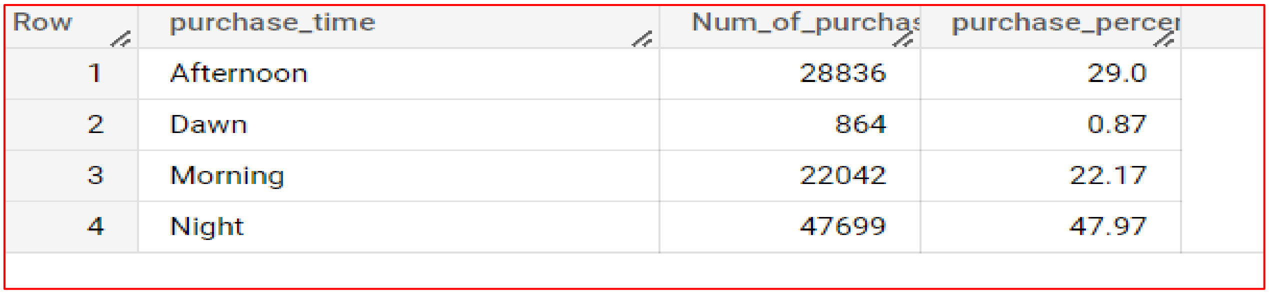


Conclusion:

1) There is an increase in number of orders of **8.99%** from 2017 to 2018 (**see Yearly growth**).
2) Data for year 2016 is given for last two quarters only. So, quarterly number of orders may provide a meaningful insight about e-commerce growing trend (**see Quarterly growth**).
3) Number of orders are increasing from quarter 3, 2016 to quarter 1,2018, while after quarter 1, 2018 orders are decreasing.
4) There is no specific month in which there is a hike in each year.

Q2 (2) What time do Brazilian customers tend to buy (Dawn, Morning, Afternoon or Night)?

```
with cte as(select purchase_time,count(purchase_time) as Num_of_purchases,sum(count(
purchase_time)) over () as total from
(select order_purchase_timestamp,
case
when extract(time from
order_purchase_timestamp) between "03:00:00" and "06:30:00"  then "Dawn"
when extract(time from order_purchase_timestamp) between "06:30:01" and "12:00:00"
 then "Morning"
when extract(time from order_purchase_timestamp) between "12:00:01" and "16:30:00"
then "Afternoon"
else "Night"
end as purchase_time
from  `target-382417.Target_sql_businesscase.orders`) as temp
group by purchase_time
order by purchase_time)
select  purchase_time, Num_of_purchases, round((Num_of_purchases/total)*100,2) as     purchase
_percentage from cte
```



Row	purchase_time	Num_of_purchas	purchase_percer
1	Afternoon	28836	29.0
2	Dawn	864	0.87
3	Morning	22042	22.17
4	Night	47699	47.97

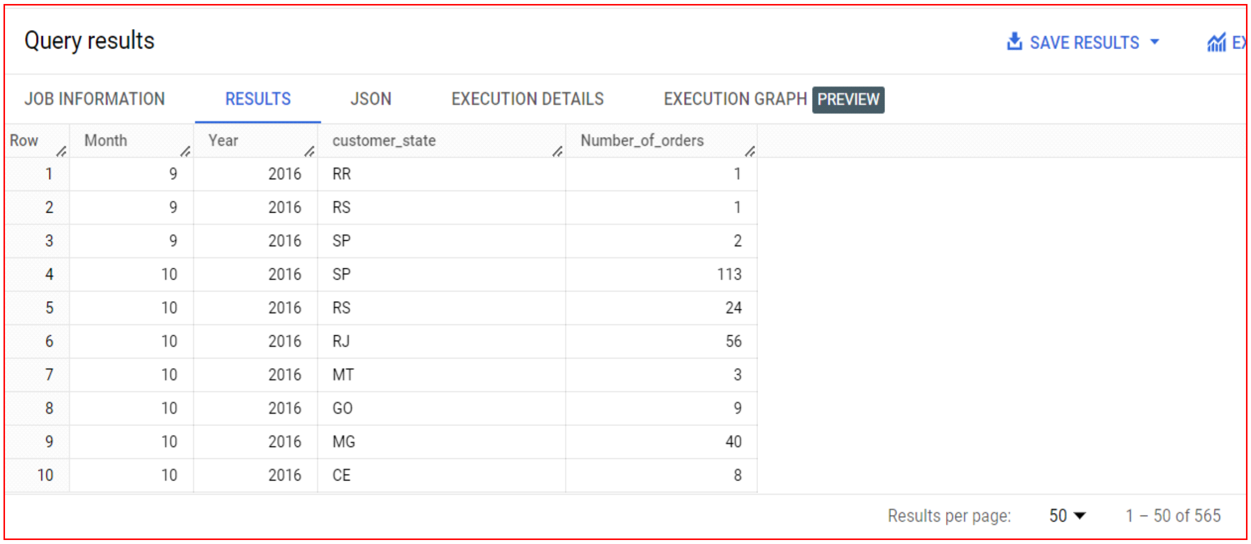
Conclusion:

1) Brazilian customers mostly do purchases in night during 16:30:01 to 2:59:59.
2) In dawn, purchase percentage is minimum.
3) They also prefer to do purchases in afternoon in comparison of morning.

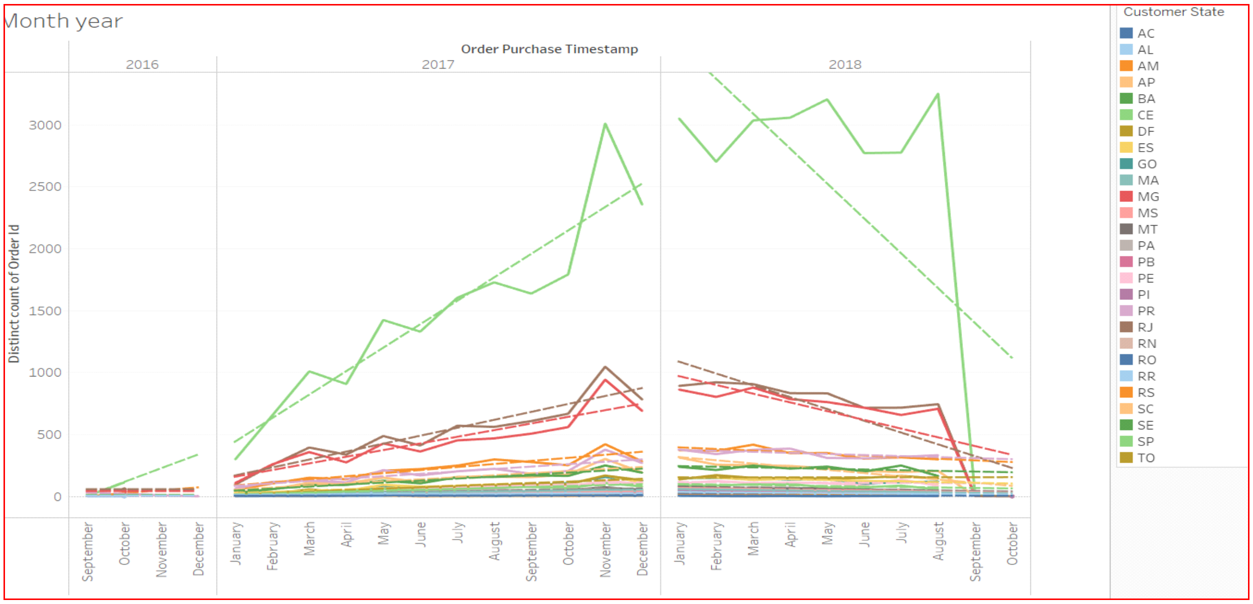
Q(3) Evolution of E-commerce orders in the Brazil region:

(1) Get month on month orders by states:

```
select  extract(MONTH FROM o.order_purchase_timestamp) AS Month,
extract(YEAR FROM o.order_purchase_timestamp) AS Year,c.customer_state,
count(distinct o.order_id) as Number_of_orders
from `target-382417.Target_sql_businesscase.orders` as o
join  `target-382417.Target_sql_businesscase.customers` as c
on c.customer_id=o.customer_id
group by Month,Year,c.customer_state
order by Year,Month
```

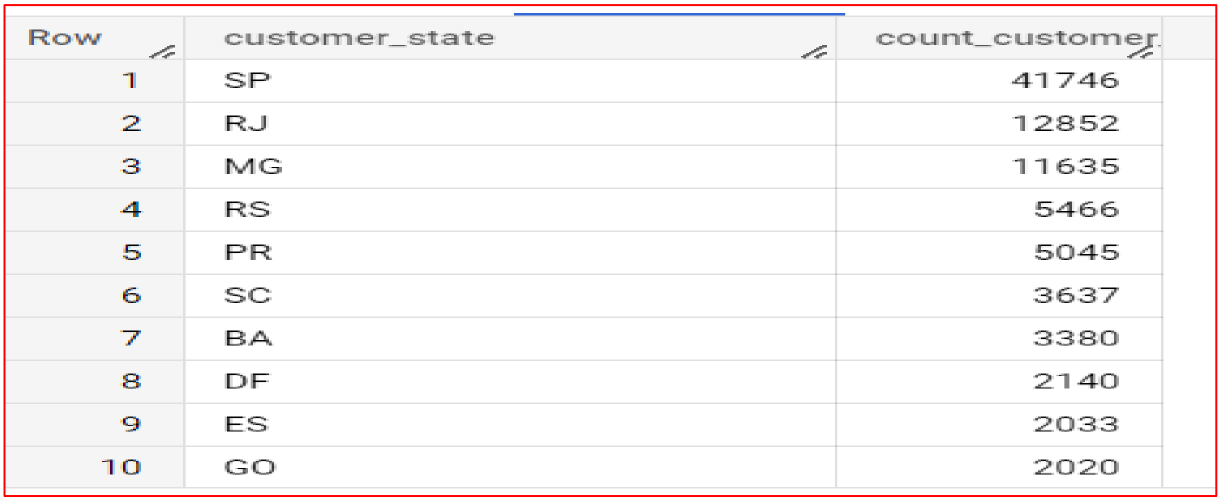


Row	Month	Year	customer_state	Number_of_orders
1	9	2016	RR	1
2	9	2016	RS	1
3	9	2016	SP	2
4	10	2016	SP	113
5	10	2016	RS	24
6	10	2016	RJ	56
7	10	2016	MT	3
8	10	2016	GO	9
9	10	2016	MG	40
10	10	2016	CE	8

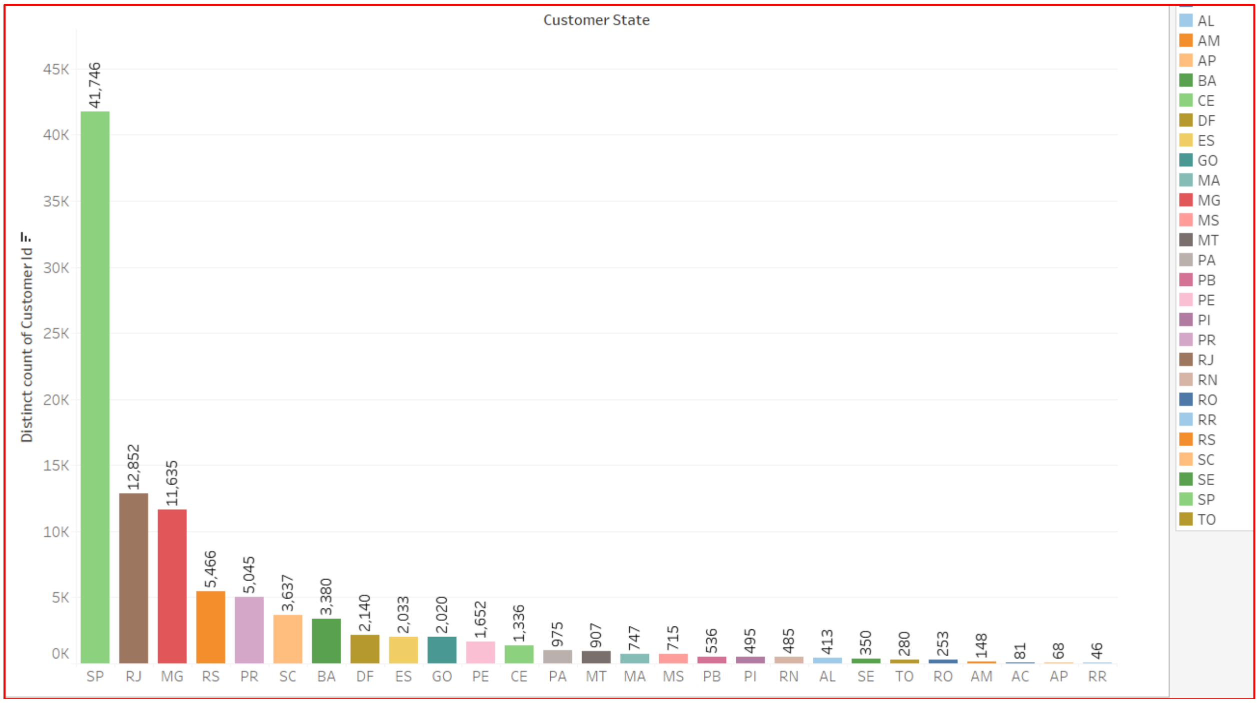


Q3(2) **Distribution of customers across the states in Brazil**

```
select  customer_state,count_customer_id
from (select count(distinct customer_id) as count_customer_id,
customer_state
from `target-382417.Target_sql_businesscase.customers`
group by customer_state) as temp
order by count_customer_id desc
```



Row	customer_state	count_customer
1	SP	41746
2	RJ	12852
3	MG	11635
4	RS	5466
5	PR	5045
6	SC	3637
7	BA	3380
8	DF	2140
9	ES	2033
10	GO	2020



Conclusions:

Q3(1):

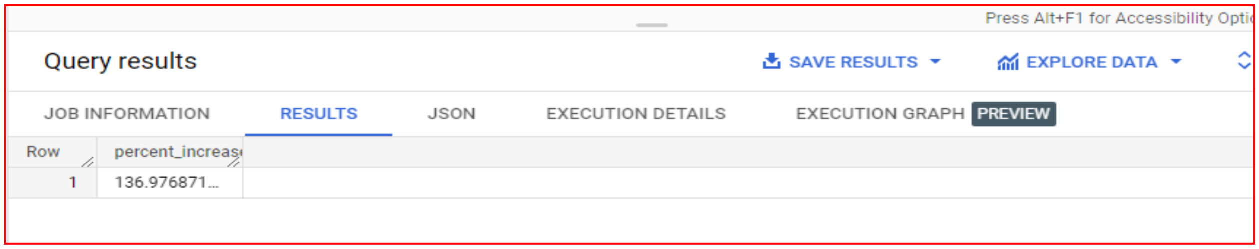
1) Just few states have growing month on month orders in 2016.
2) There is a growing trend in month on month orders for each state in 2017.
3) In 2018, the month on month orders are getting decreased for many of the states but, still orders have been increased on yearly basis as seen in Q 2 (1).

Q3(2):

1) SP and RR are the states with maximum and minimum number of customers.

Q 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

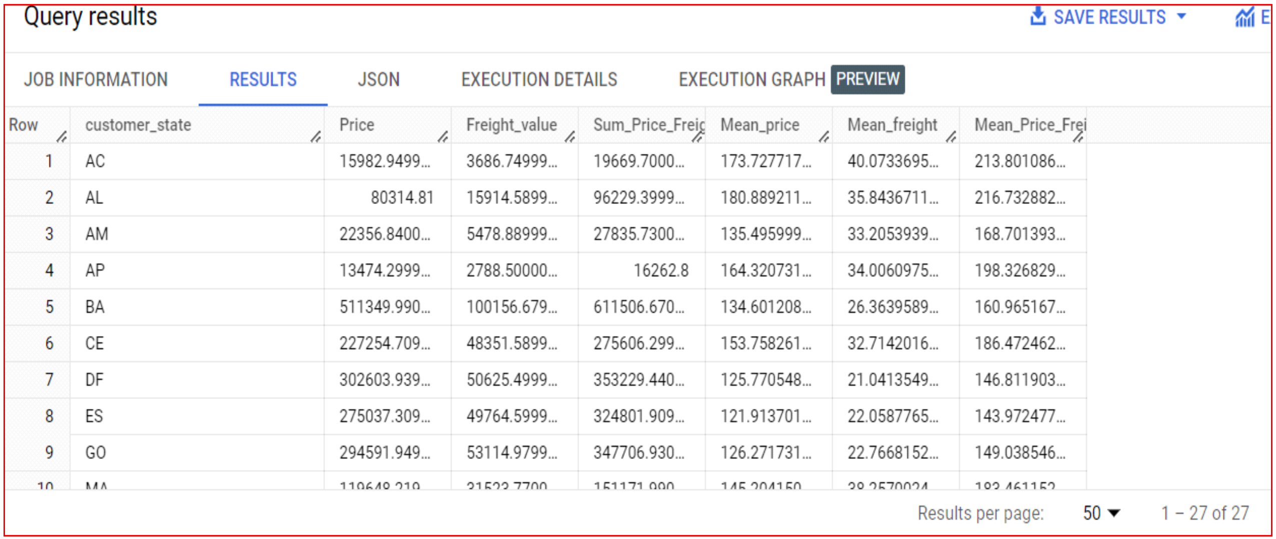
```
SELECT
    100.0 * (SUM(CASE WHEN DATE_TRUNC(o.order_purchase_timestamp, MONTH) BETWEEN '2018-01-01' AND '2018-08-31' THEN p.payment_value END)
        - SUM(CASE WHEN DATE_TRUNC(o.order_purchase_timestamp, MONTH) BETWEEN '2017-01-01' AND '2017-08-31' THEN p.payment_value END))
    / SUM(CASE WHEN DATE_TRUNC(o.order_purchase_timestamp, MONTH) BETWEEN '2017-01-01' AND '2017-08-31' THEN p.payment_value END) AS percent_increase
FROM `target-382417.Target_sql_businesscase.orders` as o
join `target-382417.Target_sql_businesscase.payments` as p
on o.order_id =p.order_id
```



Row	percent_increase
1	136.976871...

Q 4(2) Mean & Sum of price and freight value by customer state.

```
SELECT c.customer_state,sum(oi.price) as Price,sum(oi.freight_value) as Freight_value,sum(pric
e+freight_value)as Sum_Price_Freight,
AVG(oi.price) as Mean_price,AVG(oi.freight_value) as Mean_freight,
AVG(price+freight_value) as Mean_Price_Freight
from
`target-382417.Target_sql_businesscase.customers` as c
join
`target-382417.Target_sql_businesscase.orders` as o
on o.customer_id=c.customer_id
left join
`target-382417.Target_sql_businesscase.order_items` as oi
on o.order_id=oi.order_id
group by c.customer_state
```

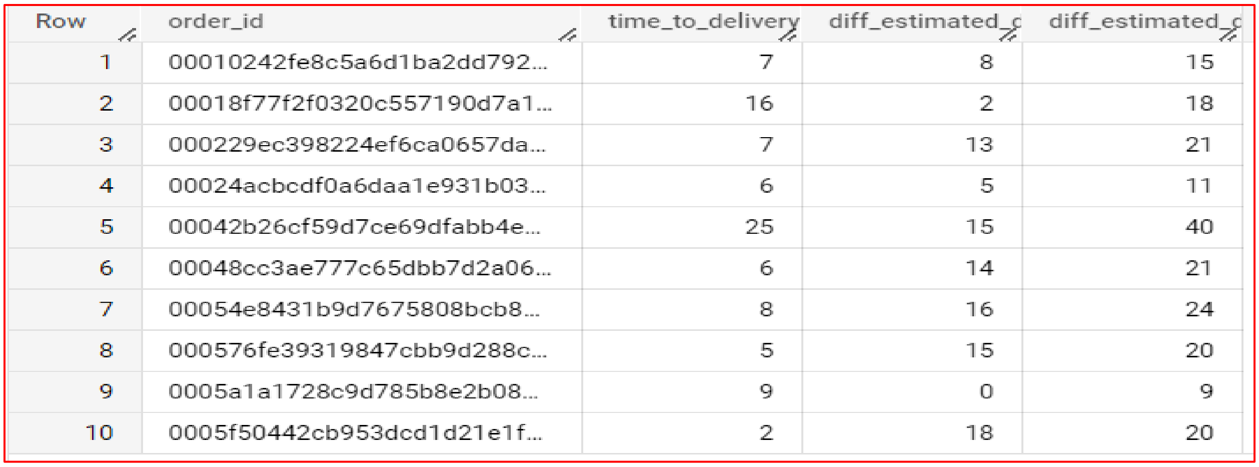


Row	customer_state	Price	Freight_value	Sum_Price_Freig	Mean_price	Mean_freight	Mean_Price_Frei
1	AC	15982.9499...	3686.74999...	19669.7000...	173.727717...	40.0733695...	213.801086...
2	AL	80314.81	15914.5899...	96229.3999...	180.889211...	35.8436711...	216.732882...
3	AM	22356.8400...	5478.88999...	27835.7300...	135.495999...	33.2053939...	168.701393...
4	AP	13474.2999...	2788.50000...	16262.8	164.320731...	34.0060975...	198.326829...
5	BA	511349.990...	100156.679...	611506.670...	134.601208...	26.3639589...	160.965167...
6	CE	227254.709...	48351.5899...	275606.299...	153.758261...	32.7142016...	186.472462...
7	DF	302603.939...	50625.4999...	353229.440...	125.770548...	21.0413549...	146.811903...
8	ES	275037.309...	49764.5999...	324801.909...	121.913701...	22.0587765...	143.972477...
9	GO	294591.949...	53114.9799...	347706.930...	126.271731...	22.7668152...	149.038546...
10	MA	119648.219	31523.7700	151171.990	145.204150	28.2570024	183.461152

Results per page: 50 ▾ 1 – 27 of 27

Q5(1) Calculate days between purchasing, delivering and estimated delivery

```
SELECT order_id,DATETIME_DIFF(order_delivered_customer_date,order_purchase_timestamp,day) as ti
me_to_delivery
DATETIME_DIFF(order_estimated_delivery_date ,order_delivered_customer_date,day)
as diff_estimated_delivery,
DATETIME_DIFF(order_estimated_delivery_date ,order_purchase_timestamp,day) as
diff_estimated_delivery_and_purchase from `target-382417.Target_sql_businesscase.orders`
order by order_id
```

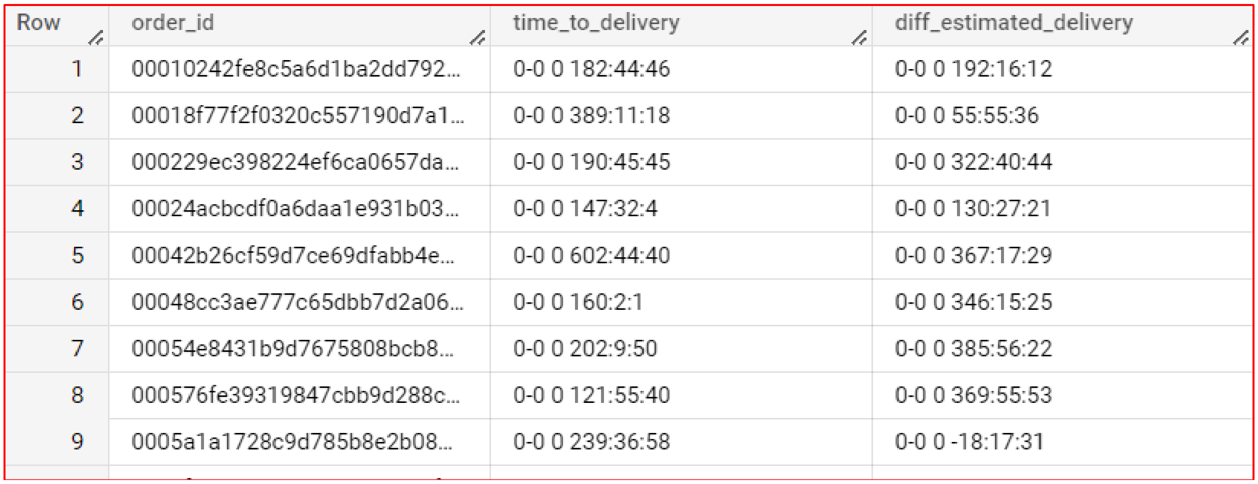


Row	order_id	time_to_delivery	diff_estimated_c	diff_estimated_c
1	00010242fe8c5a6d1ba2dd792...	7	8	15
2	00018f77f2f0320c557190d7a1...	16	2	18
3	000229ec398224ef6ca0657da...	7	13	21
4	00024acbcdf0a6daa1e931b03...	6	5	11
5	00042b26cf59d7ce69dfabb4e...	25	15	40
6	00048cc3ae777c65dbb7d2a06...	6	14	21
7	00054e8431b9d7675808bcb8...	8	16	24
8	000576fe39319847cbb9d288c...	5	15	20
9	0005a1a1728c9d785b8e2b08...	9	0	9
10	0005f50442cb953dcd1d21e1f...	2	18	20

Q 5 (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

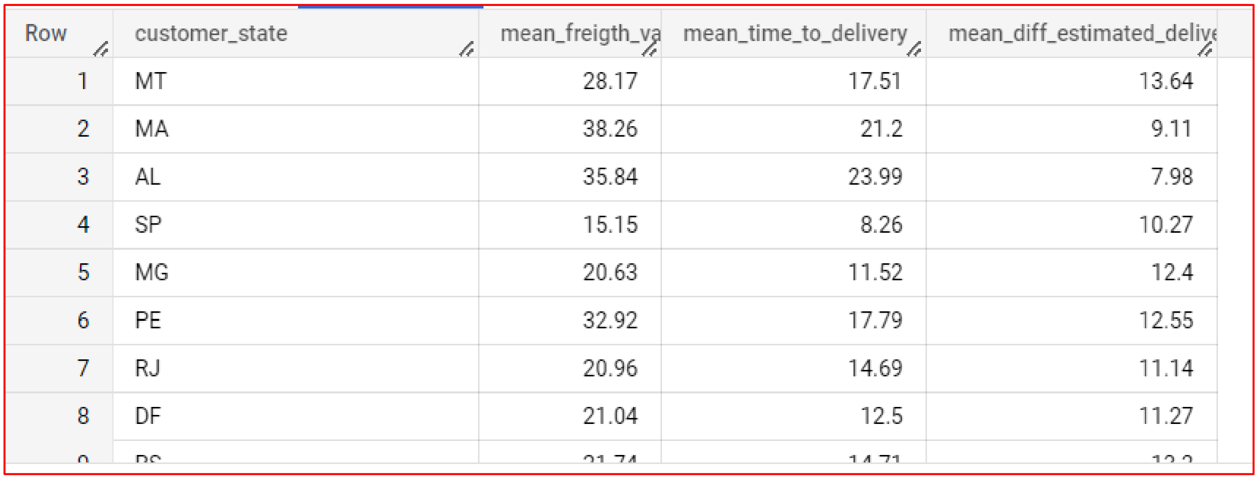
```
SELECT order_id,(order_delivered_customer_date -
 order_purchase_timestamp) as time_to_delivery,
(order_estimated_delivery_date - order_delivered_customer_date) as diff_estimated_delivery
from `target-382417.Target_sql_businesscase.orders`
order by order_id
```



Row	order_id	time_to_delivery	diff_estimated_delivery
1	00010242fe8c5a6d1ba2dd792...	0-0 0 182:44:46	0-0 0 192:16:12
2	00018f77f2f0320c557190d7a1...	0-0 0 389:11:18	0-0 0 55:55:36
3	000229ec398224ef6ca0657da...	0-0 0 190:45:45	0-0 0 322:40:44
4	00024acbcdf0a6daa1e931b03...	0-0 0 147:32:4	0-0 0 130:27:21
5	00042b26cf59d7ce69dfabb4e...	0-0 0 602:44:40	0-0 0 367:17:29
6	00048cc3ae777c65dbb7d2a06...	0-0 0 160:2:1	0-0 0 346:15:25
7	00054e8431b9d7675808bcb8...	0-0 0 202:9:50	0-0 0 385:56:22
8	000576fe39319847cbb9d288c...	0-0 0 121:55:40	0-0 0 369:55:53
9	0005a1a1728c9d785b8e2b08...	0-0 0 239:36:58	0-0 0 -18:17:31

Q5 (3) Group data by state, take mean of freight_value, time_to_delivery, diff_estimated_delivery

```
select customer_state, round(AVG(freight_value),2) as mean_freigth_value,
round(AVG(time_to_delivery),2) as mean_time_to_delivery,
round(AVG(diff_estimated_delivery),2)
as mean_diff_estimated_delivery
from
(SELECT *,DATETIME_DIFF(order_delivered_customer_date,order_purchase_timestamp,day) as time_to
_delivery,
DATETIME_DIFF(order_estimated_delivery_date  ,order_delivered_customer_date,day)
as diff_estimated_delivery
from `target-382417.Target_sql_businesscase.orders` as o
join `target-382417.Target_sql_businesscase.customers` as c
on c.customer_id=o.customer_id
join
`target-382417.Target_sql_businesscase.order_items` as oi
on oi.order_id=o.order_id
) as X
group by customer_state
```

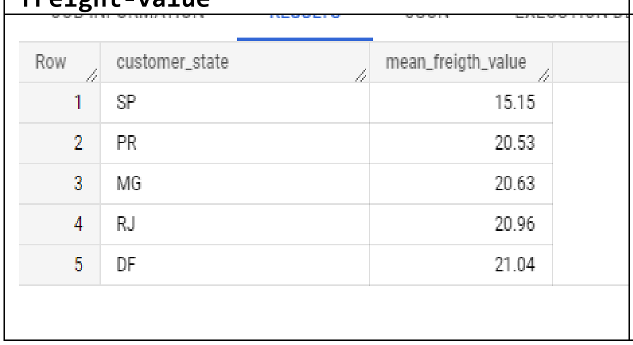


Row	customer_state	mean_freigth_va	mean_time_to_delivery	mean_diff_estimated_delive
1	MT	28.17	17.51	13.64
2	MA	38.26	21.2	9.11
3	AL	35.84	23.99	7.98
4	SP	15.15	8.26	10.27
5	MG	20.63	11.52	12.4
6	PE	32.92	17.79	12.55
7	RJ	20.96	14.69	11.14
8	DF	21.04	12.5	11.27
9	RS	21.74	14.71	13.2

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

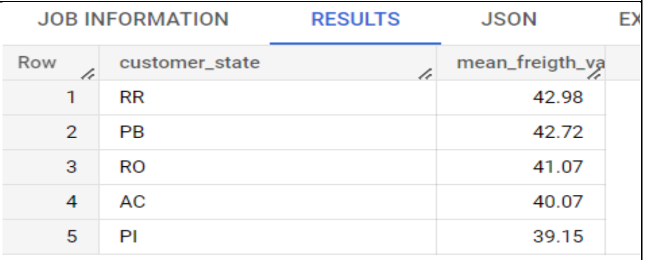
```
SELECT customer_state, round(AVG(freight_value),2) as mean_freigth_value
from `target-382417.Target_sql_businesscase.orders` as o
join `target-382417.Target_sql_businesscase.customers` as c
on c.customer_id=o.customer_id
join
`target-382417.Target_sql_businesscase.order_items` as oi  on oi.order_id=o.order_id
group by customer_state
order by mean_freigth_value
limit 5
Note: For top 5 highest average freight value, use order by mean_freigth_value desc
```

Top 5 states with lowest average freight-value



Row	customer_state	mean_freigth_value
1	SP	15.15
2	PR	20.53
3	MG	20.63
4	RJ	20.96
5	DF	21.04

Top 5 states with highest average freight-value

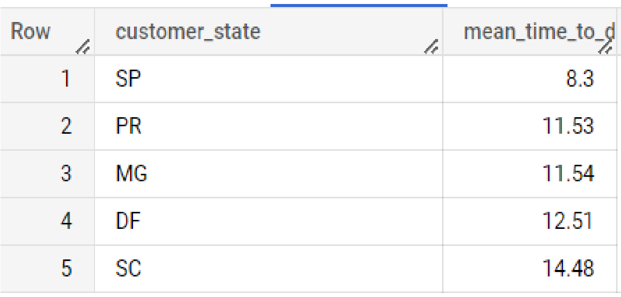


Row	customer_state	mean_freigth_va
1	RR	42.98
2	PB	42.72
3	RO	41.07
4	AC	40.07
5	PI	39.15

Q 5(6) Top 5 states with highest/lowest average time to delivery

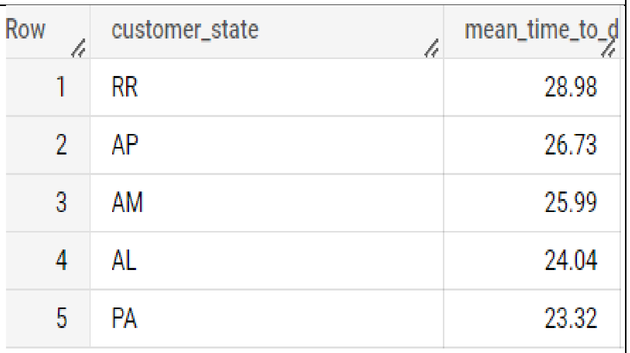
```
select customer_state, round(AVG(time_to_delivery),2) as mean_time_to_delivery,
from
(SELECT *,DATETIME_DIFF(o.order_delivered_customer_date,o.order_purchase_timestamp,day) as tim
e_to_delivery,
from `target-382417.Target_sql_businesscase.orders` as o
join `target-382417.Target_sql_businesscase.customers` as c
on c.customer_id=o.customer_id
) as X
group by customer_state
order by mean_time_to_delivery
limit 5
Note: For top 5 with highest avg time to delivery, use order by mean_time_to_delivery   desc
```

Top 5 states with lowest average time delivery



Row	customer_state	mean_time_to_d
1	SP	8.3
2	PR	11.53
3	MG	11.54
4	DF	12.51
5	SC	14.48

Top 5 states with highest average time to delivery

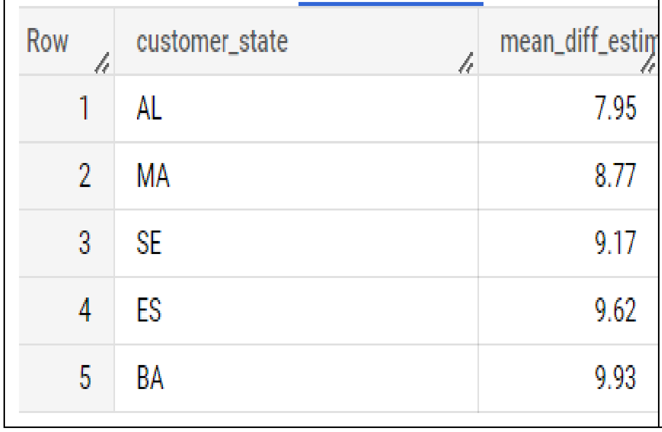


Row	customer_state	mean_time_to_d
1	RR	28.98
2	AP	26.73
3	AM	25.99
4	AL	24.04
5	PA	23.32

Q5 (7)Top 5 states where delivery is really fast/ not so fast compared to estimated date

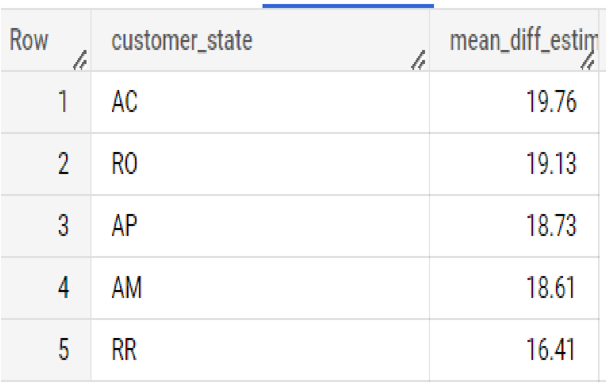
```
select customer_state, round(AVG(diff_estimated_delivery),2) as mean_diff_estimated_delive
ry,
from
(SELECT *,DATETIME_DIFF(o.order_estimated_delivery_date ,o.order_delivered_customer_date,d
ay)
as diff_estimated_delivery,
from `target-382417.Target_sql_businesscase.orders` as o
join `target-382417.Target_sql_businesscase.customers` as c
on c.customer_id=o.customer_id
) as X
group by customer_state
order by mean_diff_estimated_delivery
limit 5
Note: For Top 5, use order by mean_diff_estimated_delivery desc
```

Top 5 States with slowest delivery



Row	customer_state	mean_diff_estim
1	AL	7.95
2	MA	8.77
3	SE	9.17
4	ES	9.62
5	BA	9.93

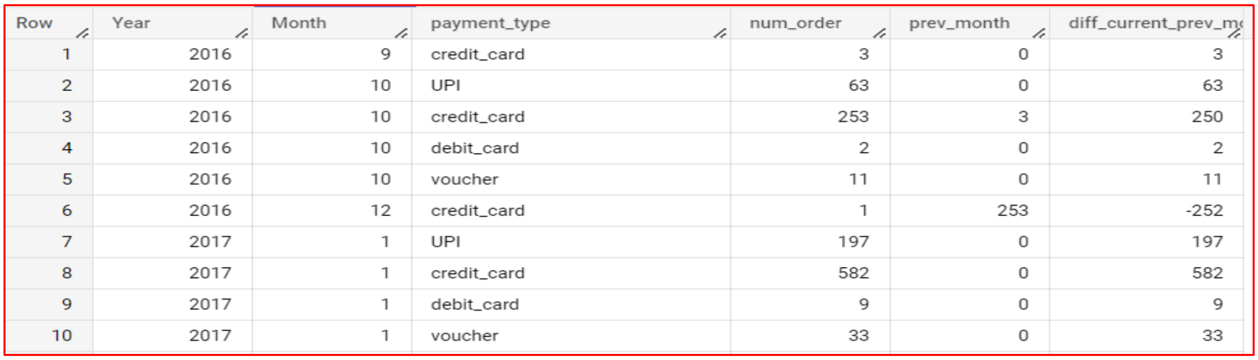
Top 5 States with fastest delivery



Row	customer_state	mean_diff_estim
1	AC	19.76
2	RO	19.13
3	AP	18.73
4	AM	18.61
5	RR	16.41

Q 6(1) Month over Month count of orders for different payment types

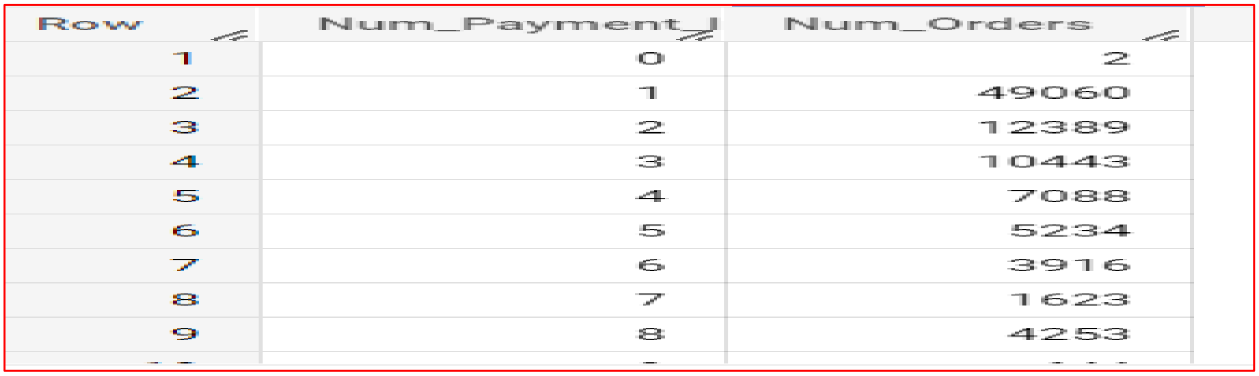
```
select *,Lag(num_order,1,0) over (partition by payment_type,Year order by Month)
as prev_month_order,
(num_order-Lag(num_order,1,0) over (partition by payment_type,Year order by Month))
as diff_current_prev_month,
from (select extract(Year from o.order_purchase_timestamp) AS Year,
extract(Month from o.order_purchase_timestamp) AS Month,   p.payment_type,
count(distinct o.order_id) as num_order from
`target-382417.Target_sql_businesscase.orders` as o
join `target-382417.Target_sql_businesscase.payments` as p
on o.order_id=p.order_id
group by Year,Month,p.payment_type) as temp
order by Year,Month,payment_type
```



Row	Year	Month	payment_type	num_order	prev_month	diff_current_prev_m
1	2016	9	credit_card	3	0	3
2	2016	10	UPI	63	0	63
3	2016	10	credit_card	253	3	250
4	2016	10	debit_card	2	0	2
5	2016	10	voucher	11	0	11
6	2016	12	credit_card	1	253	-252
7	2017	1	UPI	197	0	197
8	2017	1	credit_card	582	0	582
9	2017	1	debit_card	9	0	9
10	2017	1	voucher	33	0	33

Q. 6(2) Count of orders based on the no. of payment installments

```
Select  payment_installments as Num_Payment_Installments, count (distinct order_id) as Num_Ord
ers
from `target-382417.Target_sql_businesscase.payments`
group by Num_Payment_Installments
order by Num_Payment_Installments
```

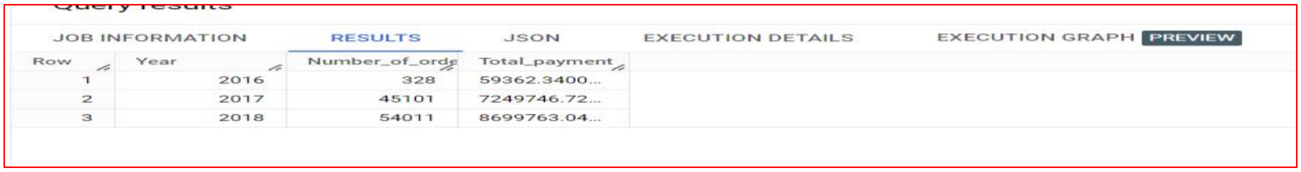


Row	Num_Payment_I	Num_Orders
1	0	2
2	1	49060
3	2	12389
4	3	10443
5	4	7088
6	5	5234
7	6	3916
8	7	1623
9	8	4253

Q(7) Insights: (1)<u>Payment due for an order</u>

(i)Yearwise total number of orders for which payments are received.

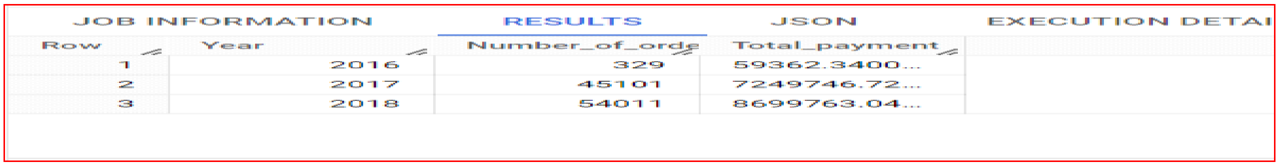
```
select  extract(YEAR FROM o.order_purchase_timestamp) AS Year,
count(distinct o.order_id) as Number_of_orders,
sum(p.payment_value) as Total_payment,
from `target-382417.Target_sql_businesscase.orders` as o
join  `target-382417.Target_sql_businesscase.payments`as p
on o.order_id=p.order_id
group by Year
order by Year
```



Row	Year	Number_of_orde	Total_payment
1	2016	328	59362.3400...
2	2017	45101	7249746.72...
3	2018	54011	8699763.04...

(ii) Year-wise total number of orders and payments received.

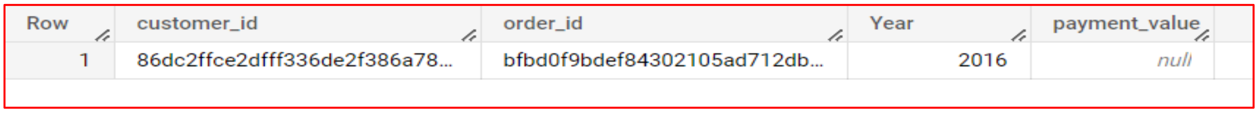
```
select  extract(YEAR FROM o.order_purchase_timestamp) AS Year,
count(distinct o.order_id) as Number_of_orders,
sum(p.payment_value) as Total_payment,
from `target-382417.Target_sql_businesscase.orders` as o
left join  `target-382417.Target_sql_businesscase.payments`as p
on o.order_id=p.order_id
group by Year
order by Year
```



Row	Year	Number_of_orde	Total_payment
1	2016	329	59362.3400...
2	2017	45101	7249746.72...
3	2018	54011	8699763.04...

(iii) Orders for which payments are not received.

```
select c.customer_id, o.order_id, o.order_purchase_timestamp,
p.payment_value
from `target-382417.Target_sql_businesscase.orders` as o
left join  `target-382417.Target_sql_businesscase.payments`as p
on o.order_id=p.order_id
join `target-382417.Target_sql_businesscase.customers` c
on c.customer_id=o.customer_id
where p.payment_value is null
```



Row	customer_id	order_id	Year	payment_value
1	86dc2ffce2dfff336de2f386a78...	bfbd0f9bdef84302105ad712db...	2016	null

Conclusions:

1) Number of orders in 2016 are 328 in (i) whereas in (ii) number of orders are 329. There is an order for which payment is not received.
2) The order_id is given in (iii).

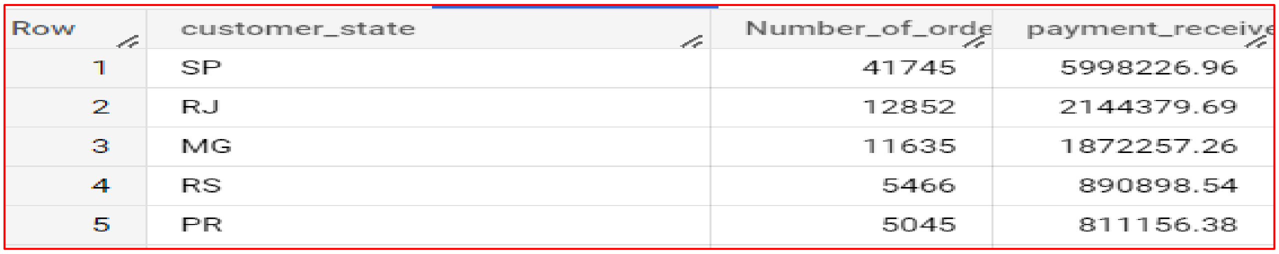
(2) Insight: Customers preferable time to buy from Q2

1) Brazilian customers mostly do purchases in night during 16:30:00 to 2:59:59.
2) In dawn, purchase percentage is minimum.
3) They also prefer to do purchases in afternoon in comparison of morning.

(3) Insight: Top 5 states with maximum and minimum sales

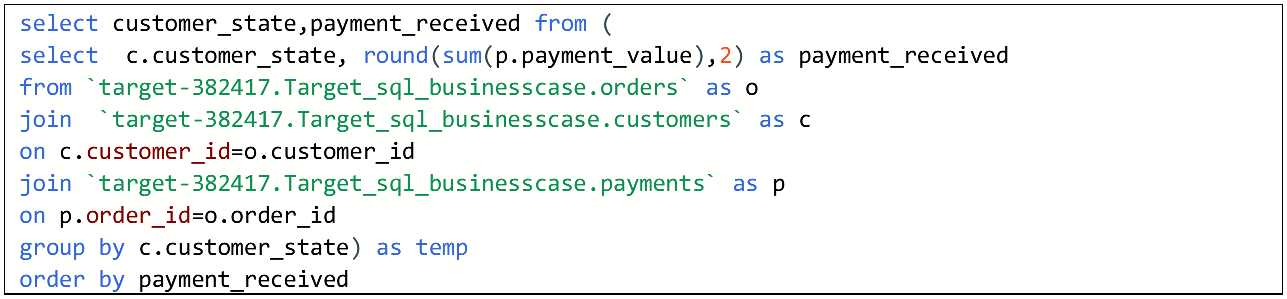
Top 5 States with maximum orders and sales

```
 select customer_state,Number_of_orders,payment_received from (
select  c.customer_state,count(distinct o.order_id) as Number_of_orders,round(sum(p.payment_va
lue),2) as payment_received
from `target-382417.Target_sql_businesscase.orders` as o
join  `target-382417.Target_sql_businesscase.customers` as c
on c.customer_id=o.customer_id
join `target-382417.Target_sql_businesscase.payments` as p
on p.order_id=o.order_id
group by c.customer_state) as temp
order by payment_received desc
```

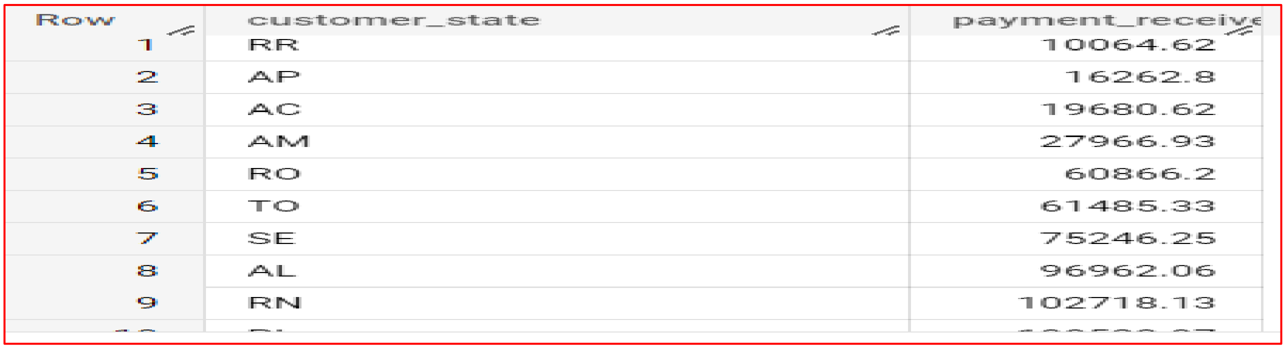


Row	customer_state	Number_of_orde	payment_receive
1	SP	41745	5998226.96
2	RJ	12852	2144379.69
3	MG	11635	1872257.26
4	RS	5466	890898.54
5	PR	5045	811156.38

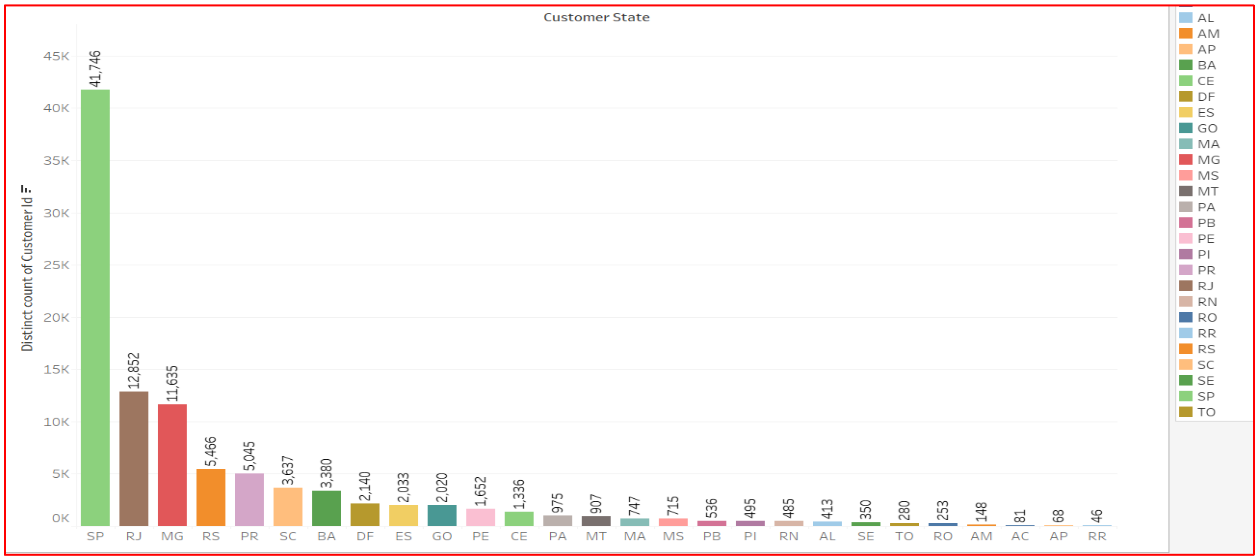
Top 5 States with minimum sales:



```
select customer_state,payment_received from (
select  c.customer_state, round(sum(p.payment_value),2) as payment_received
from `target-382417.Target_sql_businesscase.orders` as o
join  `target-382417.Target_sql_businesscase.customers` as c
on c.customer_id=o.customer_id
join `target-382417.Target_sql_businesscase.payments` as p
on p.order_id=o.order_id
group by c.customer_state) as temp
order by payment_received
```



Row	customer_state	payment_receive
1	RR	10064.62
2	AP	16262.8
3	AC	19680.62
4	AM	27966.93
5	RO	60866.2
6	TO	61485.33
7	SE	75246.25
8	AL	96962.06
9	RN	102718.13

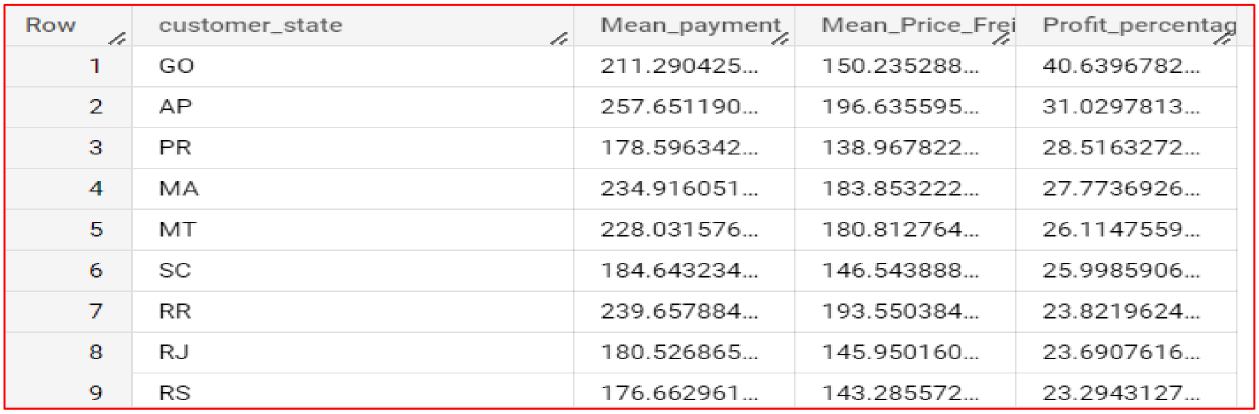


Observations:

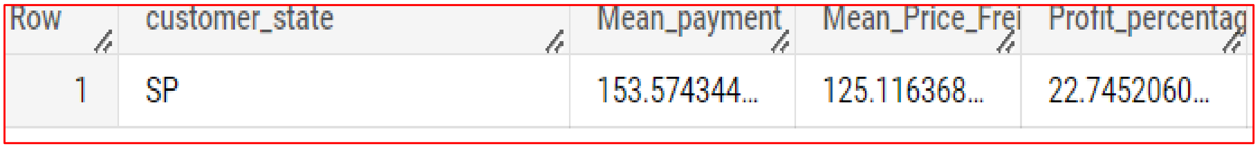
1) SP,RJ,MG,RS and PR are top 5 states with maximum number of customers, orders and payments.
2) RR and AP are states with minimum sales.

4) Average profit percentage:

```
select customer_state,Mean_payment,Mean_Price_Freight,((Mean_payment-
Mean_Price_Freight)/Mean_Price_Freight)*100 as Profit_percentage from
(SELECT c.customer_state,AVG(oi.price) as Mean_price,AVG(oi.freight_value) as Mean_freight,Avg
(p.payment_value) as Mean_payment,
AVG(price+freight_value) as Mean_Price_Freight
from
`target-382417.Target_sql_businesscase.customers` as c
join
`target-382417.Target_sql_businesscase.orders` as o
on o.customer_id=c.customer_id
left join
`target-382417.Target_sql_businesscase.order_items` as oi
on o.order_id=oi.order_id
join `target-382417.Target_sql_businesscase.payments` as p
on o.order_id=p.order_id
group by c.customer_state
order by Mean_Price_Freight desc) as temp
order by Profit_percentage desc
**add where c.customer_state=”SP” to find SP profit percentage before group by
```



Row	customer_state	Mean_payment	Mean_Price_Frei	Profit_percentag
1	GO	211.290425...	150.235288...	40.6396782...
2	AP	257.651190...	196.635595...	31.0297813...
3	PR	178.596342...	138.967822...	28.5163272...
4	MA	234.916051...	183.853222...	27.7736926...
5	MT	228.031576...	180.812764...	26.1147559...
6	SC	184.643234...	146.543888...	25.9985906...
7	RR	239.657884...	193.550384...	23.8219624...
8	RJ	180.526865...	145.950160...	23.6907616...
9	RS	176.662961...	143.285572...	23.2943127...



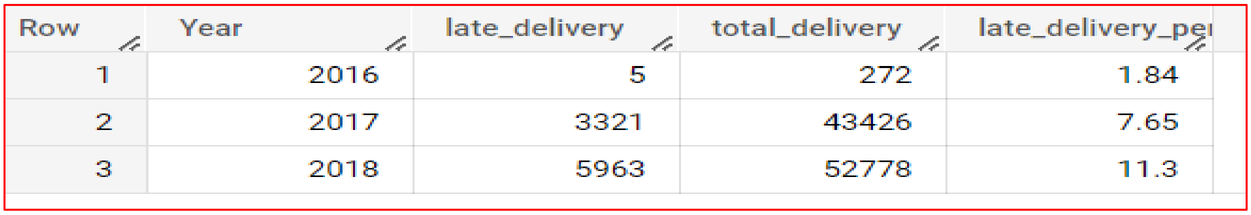
Row	customer_state	Mean_payment	Mean_Price_Frei	Profit_percentag
1	SP	153.574344...	125.116368...	22.7452060...

Observations:

1) RR and AP have minimum sale but their profit percentage are appreciable.
2) Go is in top 10 maximum sale and has maximum profit percentage so could be a choice for new outlet opening.
3) SP is the state with maximum sale but profit percentage is not in the list of top 10.

5) **Insight: Increasing late delivery percentage**

```
select Year,sum(late_deliveryy) as late_delivery,
count(Days_between_estimated_delivery_delivery) as total_delivery,
round((sum(late_deliveryy)/count(Days_between_estimated_delivery_delivery))*100,2) as late_delivery_percentage
from(SELECT DATETIME_DIFF(order_estimated_delivery_date  ,order_delivered_customer_date,day)
as Days_between_estimated_delivery_delivery, extract(YEAR FROM order_purchase_timestamp) as Year,
case
when DATETIME_DIFF(order_estimated_delivery_date  ,order_delivered_customer_date,day) >0 or
DATETIME_DIFF(order_estimated_delivery_date  ,order_delivered_customer_date,day) is
 null  then 0
else 1
end as late_deliveryy
from `target-382417.Target_sql_businesscase.orders`
order by Year) as temp
group by Year
order by Year
```

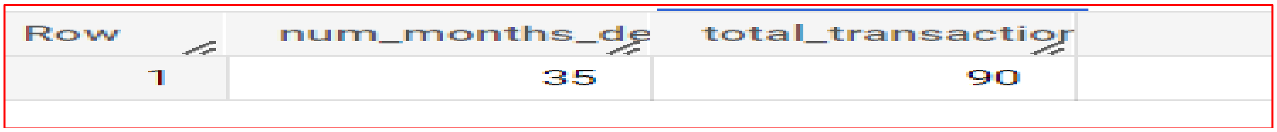


Row	Year	late_delivery	total_delivery	late_delivery_pe
1	2016	5	272	1.84
2	2017	3321	43426	7.65
3	2018	5963	52778	11.3

(6) Insight: Payment type analysis

Payment type	Total month of transactions	Count of Months with decreasing orders
Credit card	**23**	**8**
Debit card	**21**	**6**
UPI	**21**	**9**
Voucher	**23**	**21**
Not defined	**2**	**1**
Total	**90**	**35**

```
with cte as (select payment_type,case when (diff_current_prev_month) < 0 then 1
else 0
end as decreasing_order_count from (select *,Lag(num_order,1,0) over (partition by payment_typ
e,Year order by Month) as prev_month,
(num_order-
Lag(num_order,1,0) over (partition by payment_type,Year order by Month)) as diff_current_prev_
month,
from (select extract(Year from o.order_purchase_timestamp) AS Year,
extract(Month from o.order_purchase_timestamp) AS Month,   p.payment_type,
count(distinct o.order_id) as num_order from
`target-382417.Target_sql_businesscase.orders` as o
join `target-382417.Target_sql_businesscase.payments` as p
on o.order_id=p.order_id
group by Year,Month,p.payment_type) as temp
order by Year,Month,payment_type) X
order by Year,Month,payment_type)

select sum(decreasing_order_count) as num_months_decreasing_order,count(decreasing_order_count
) total_transaction_month from cte
Note: Use where payment_type = "name of payment type" to find data for each payment type
```



Row	num_months_de	total_transactior
1	35	90

Q (8) (1) Recommendations:

1) Do a follow up with a reminder to the customer for the payment due.(Insight 1)
2) Target can scale up its operations in Brazil as number of orders are getting increased yearly. (Q2(1))
3) Offer a referral program, discounts to make more purchases or a refund guarantee in the Top 5 states with minimum sale, especially in AP as it has 2nd highest profit percentage.(Insight 3 and 4)
4) Offer happy hours during dawn to attract more customers.(Q2)
5) RR,AP and GO may be better choices to open new outlets in terms of profit percentage. (Insight 4)
6) Tend to minimize the increasing percentage of delay in delivery services. (Insight 5)
7) Payment via voucher is decreasing almost each month, so Target may discontinue with voucher payment mode.(Insight 6)