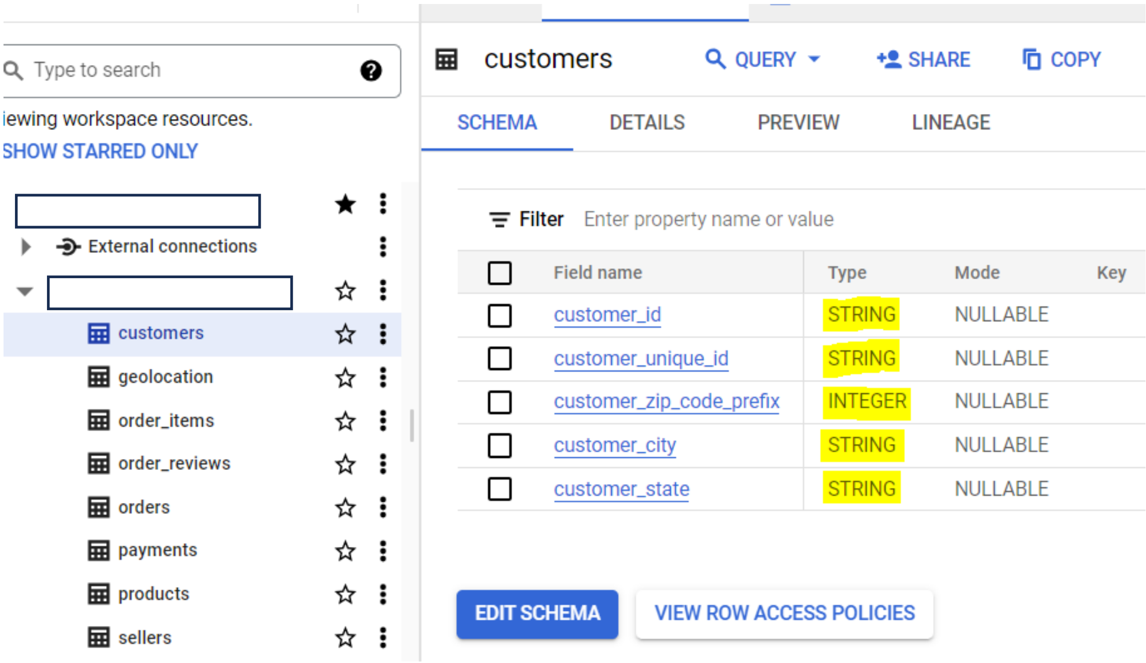
US based Global retailers Tableau & SQL Business Case Study (Akshata Gupta)

Contents

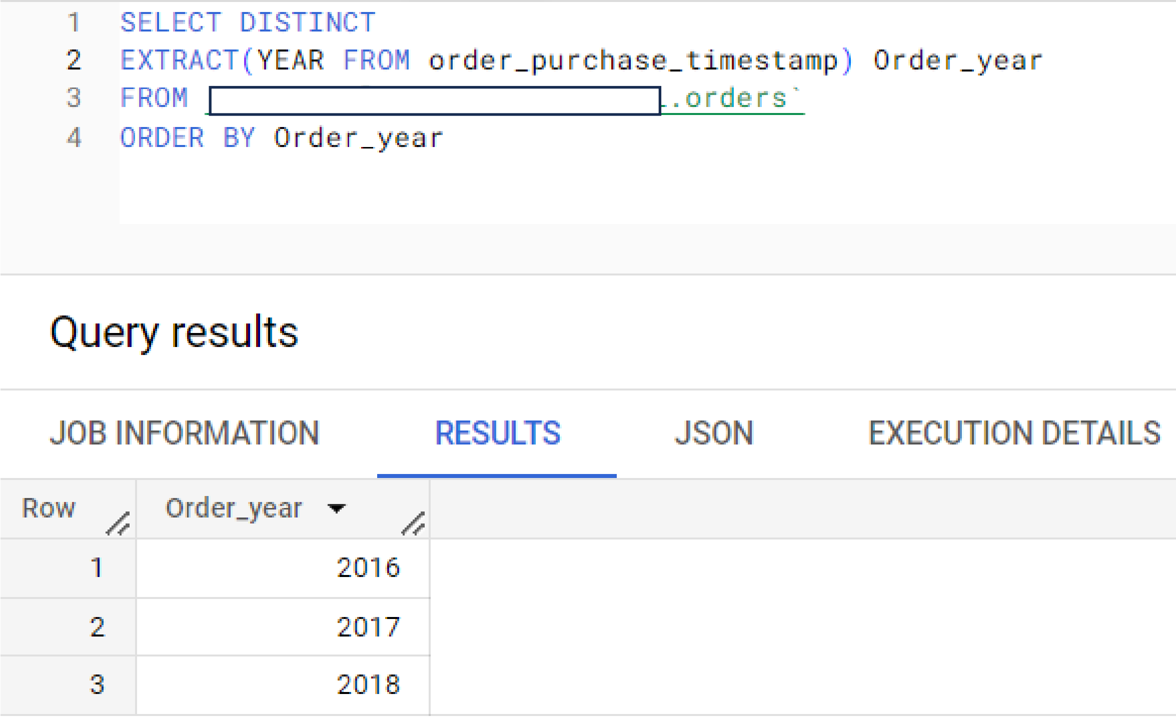
Exploratory Analysis

- Data type of all columns in the "customers" table.



Field name	Type	Mode	Key
customer_id	STRING	NULLABLE	
customer_unique_id	STRING	NULLABLE	
customer_zip_code_prefix	INTEGER	NULLABLE	
customer_city	STRING	NULLABLE	
customer_state	STRING	NULLABLE	

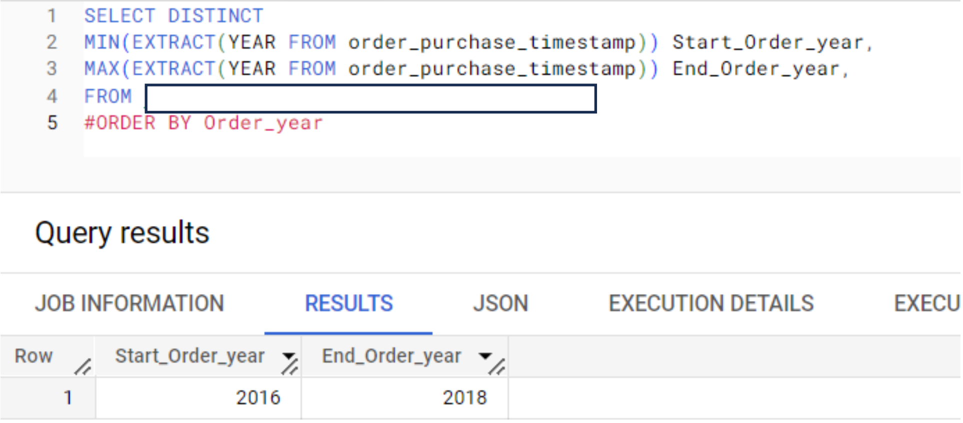
- Get the time range between which the orders were placed.



```sql
SELECT DISTINCT
EXTRACT(YEAR FROM order_purchase_timestamp) Order_year
FROM [redacted].orders`
ORDER BY Order_year
```

Query results

Row	Order_year
1	2016
2	2017
3	2018

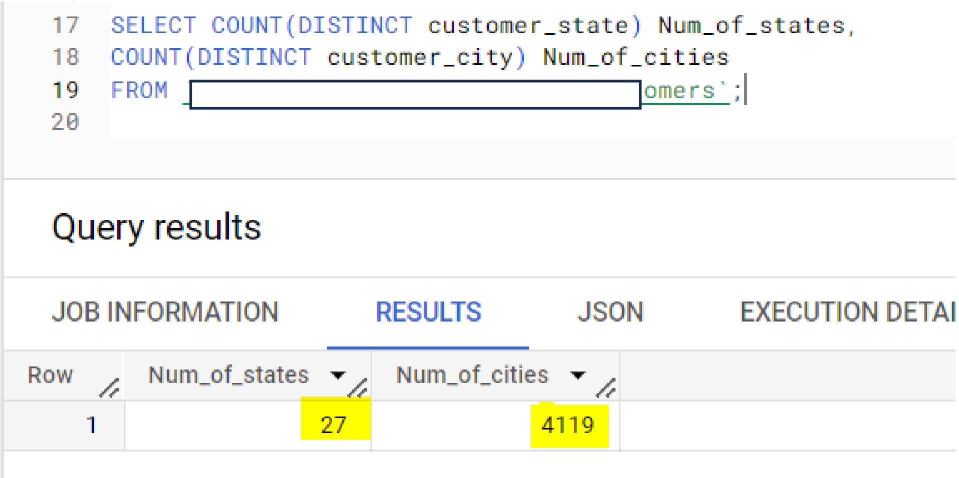


```sql
SELECT DISTINCT
MIN(EXTRACT(YEAR FROM order_purchase_timestamp)) Start_Order_year,
MAX(EXTRACT(YEAR FROM order_purchase_timestamp)) End_Order_year,
FROM [redacted]
#ORDER BY Order_year
```

Query results

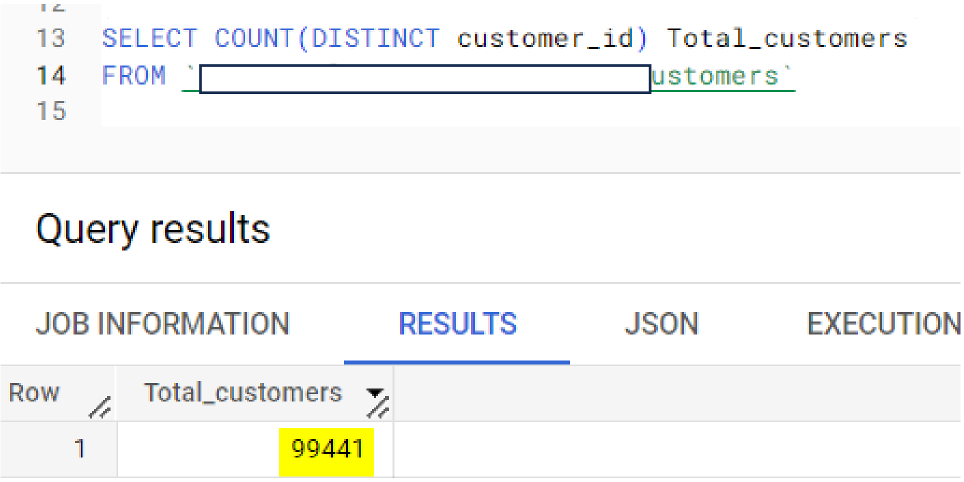
Row	Start_Order_year	End_Order_year
1	2016	2018

- ➢ Count the Cities & States of customers who ordered during the given period.



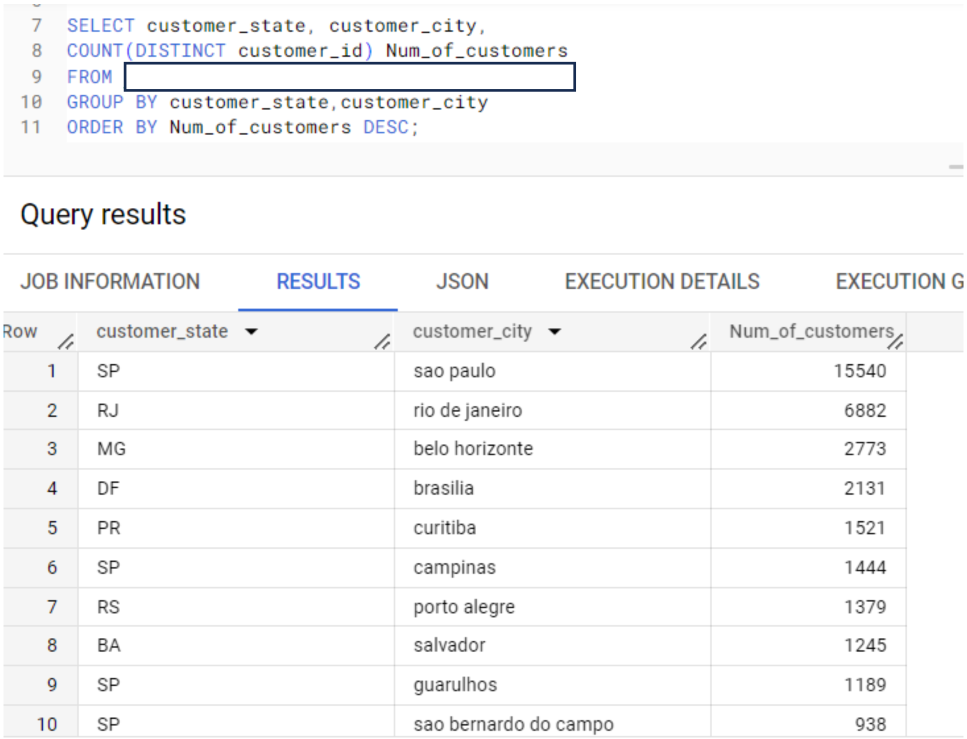
```
SELECT COUNT(DISTINCT customer_state) Num_of_states,
COUNT(DISTINCT customer_city) Num_of_cities
FROM [redacted]omers`;
```

Row	Num_of_states	Num_of_cities
1	27	4119



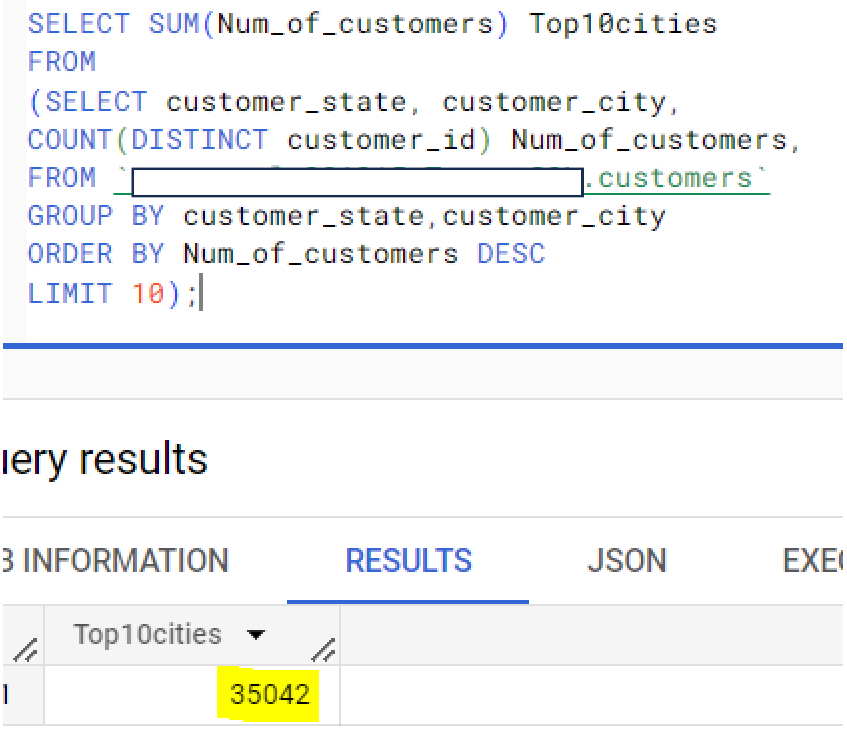
```
SELECT COUNT(DISTINCT customer_id) Total_customers
FROM `[redacted]ustomers`
```

Row	Total_customers
1	99441



```
SELECT customer_state, customer_city,
COUNT(DISTINCT customer_id) Num_of_customers
FROM [redacted]
GROUP BY customer_state,customer_city
ORDER BY Num_of_customers DESC;
```

Row	customer_state	customer_city	Num_of_customers
1	SP	sao paulo	15540
2	RJ	rio de janeiro	6882
3	MG	belo horizonte	2773
4	DF	brasilia	2131
5	PR	curitiba	1521
6	SP	campinas	1444
7	RS	porto alegre	1379
8	BA	salvador	1245
9	SP	guarulhos	1189
10	SP	sao bernardo do campo	938



```
SELECT SUM(Num_of_customers) Top10cities
FROM
(SELECT customer_state, customer_city,
COUNT(DISTINCT customer_id) Num_of_customers,
FROM `[redacted].customers`
GROUP BY customer_state,customer_city
ORDER BY Num_of_customers DESC
LIMIT 10);
```

	Top10cities
1	35042

Insights:

- The dataset available is of customers’ orders placed from 2016 to 2018
- Customers from over 27 different states & 4119 different cities in Brazil have placed these orders

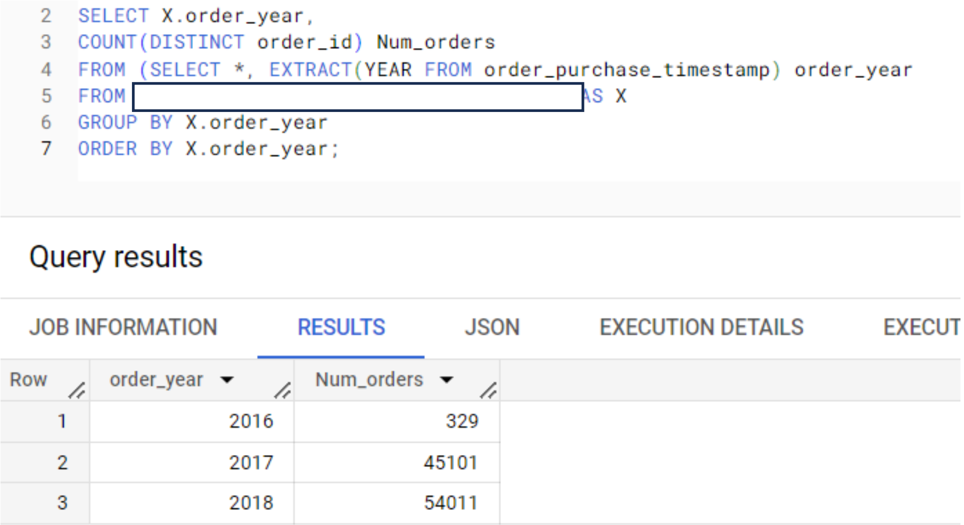
- A total of 99,441 customers’ have placed orders in Brazil operations of XYZ Co. during this period.
- Over 15.62% (15540 / 99441) customers of XYZ Co. in Brazil are from Sao Paulo city.
- Over 35.23% (35042 / 99441) customers are in the Top 10 cities alone.

Action Items:

- Checking the sales, profit, etc. Parameters for customers in Sao Paulo city and other Top 10 cities would help cater to majority of the XYZ Co. Brazil customers and have impactful outcomes.

In-depth Exploration:

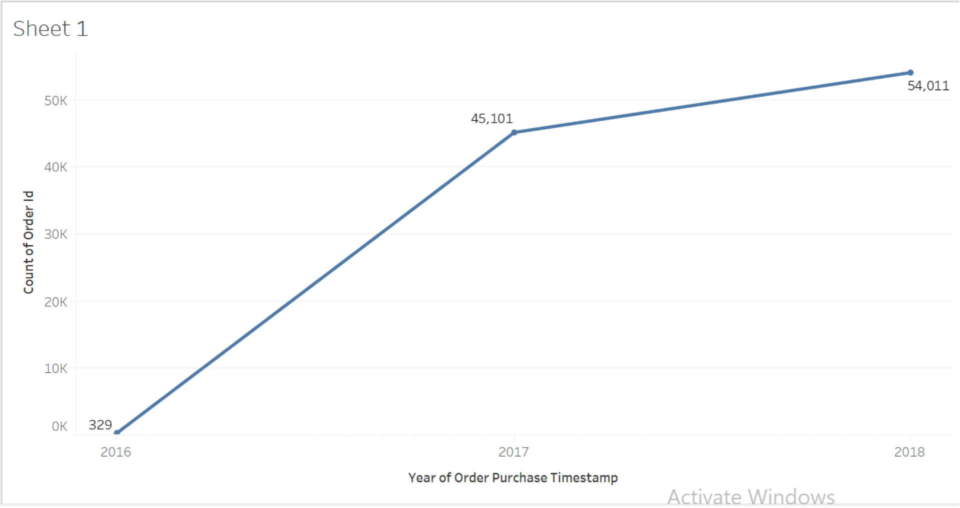
- ➢ Is there a growing trend in the no. of orders placed over the past years?



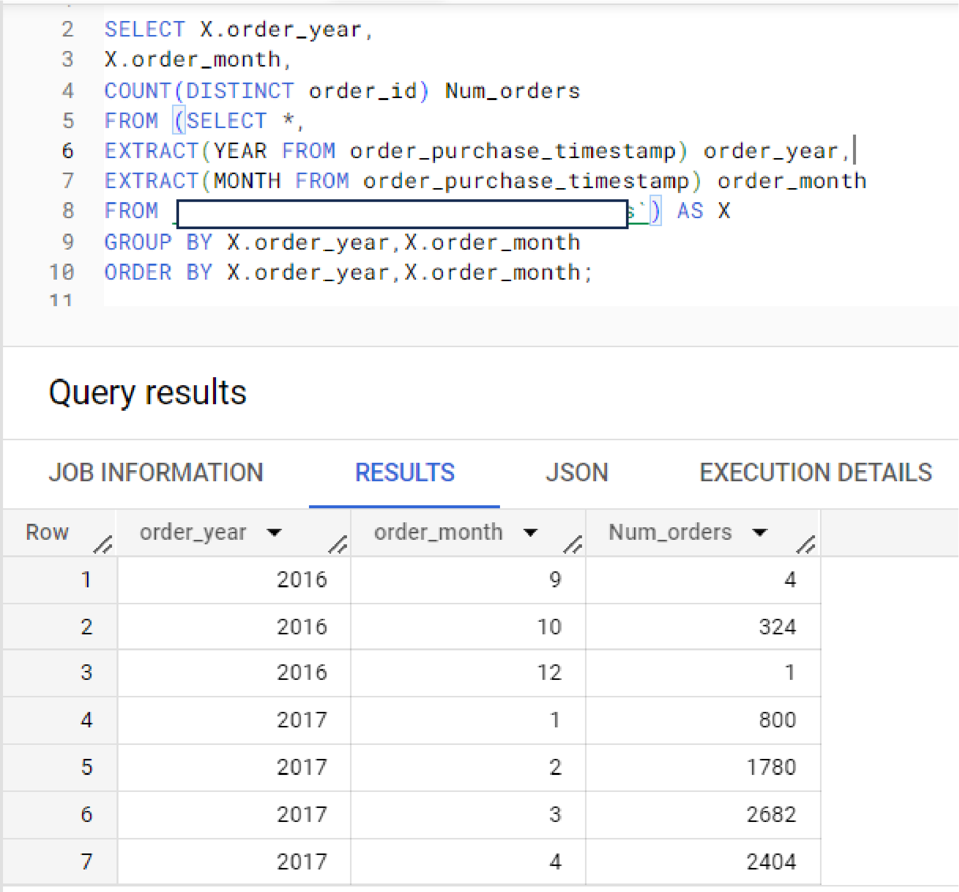
```
SELECT X.order_year,
COUNT(DISTINCT order_id) Num_orders
FROM (SELECT *, EXTRACT(YEAR FROM order_purchase_timestamp) order_year
FROM [redacted] AS X
GROUP BY X.order_year
ORDER BY X.order_year;
```

Query results

Row	order_year	Num_orders
1	2016	329
2	2017	45101
3	2018	54011



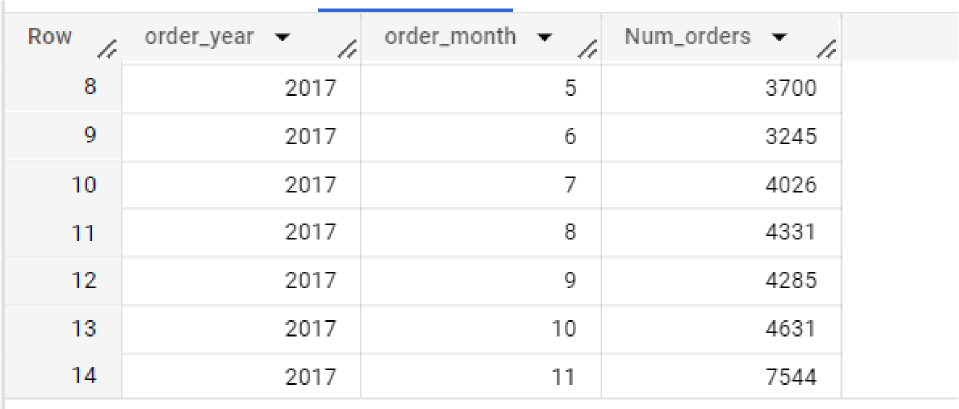
- ➢ Can we see some kind of monthly seasonality in terms of the no. of orders being placed?



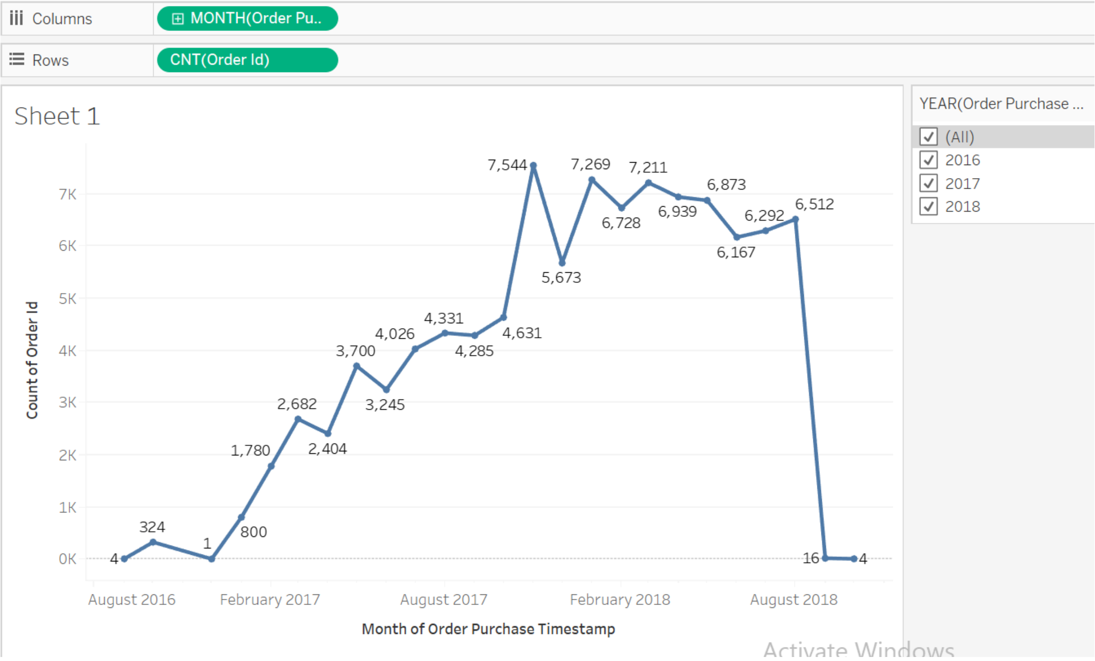
```
SELECT X.order_year,
X.order_month,
COUNT(DISTINCT order_id) Num_orders
FROM (SELECT *,
EXTRACT(YEAR FROM order_purchase_timestamp) order_year,
EXTRACT(MONTH FROM order_purchase_timestamp) order_month
FROM [redacted]) AS X
GROUP BY X.order_year,X.order_month
ORDER BY X.order_year,X.order_month;
```

Query results

Row	order_year	order_month	Num_orders
1	2016	9	4
2	2016	10	324
3	2016	12	1
4	2017	1	800
5	2017	2	1780
6	2017	3	2682
7	2017	4	2404



Row	order_year	order_month	Num_orders
8	2017	5	3700
9	2017	6	3245
10	2017	7	4026
11	2017	8	4331
12	2017	9	4285
13	2017	10	4631
14	2017	11	7544



Columns: MONTH(Order Pu..

Rows: CNT(Order Id)

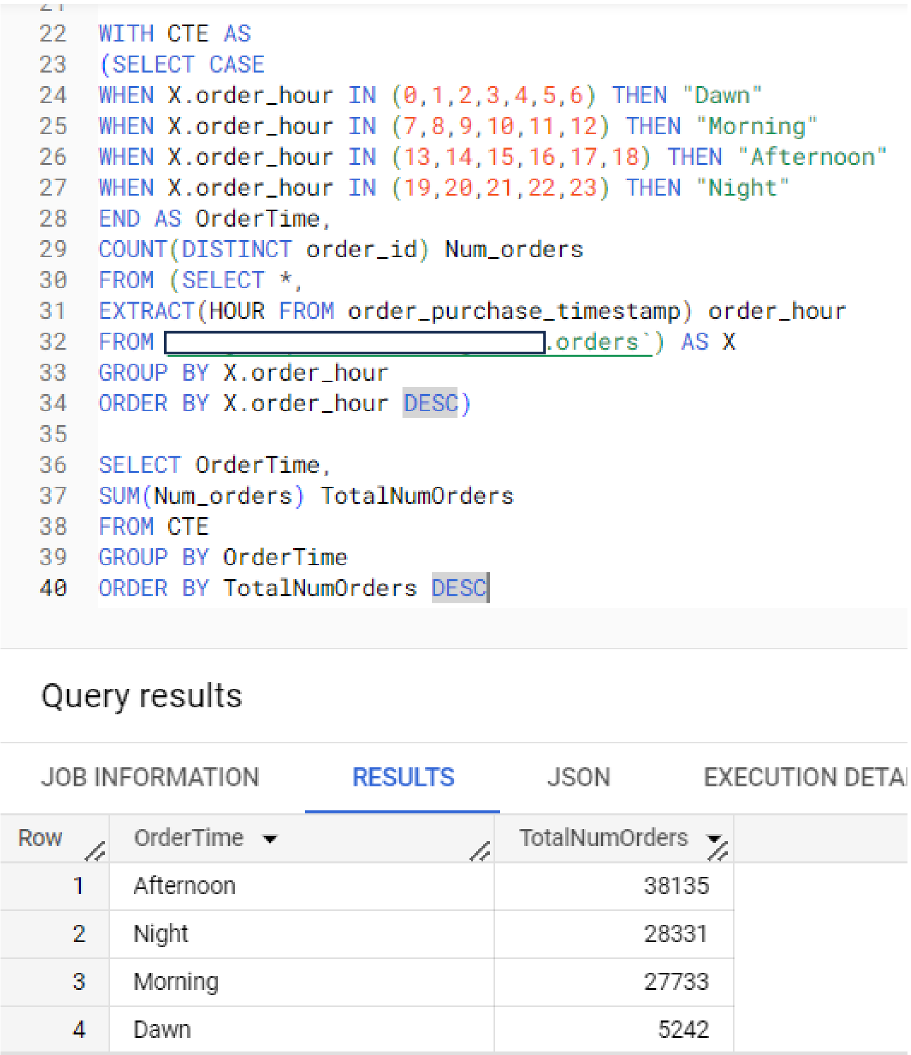
Sheet 1

YEAR(Order Purchase ...

☑ (All)
☑ 2016
☑ 2017
☑ 2018

➢ 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

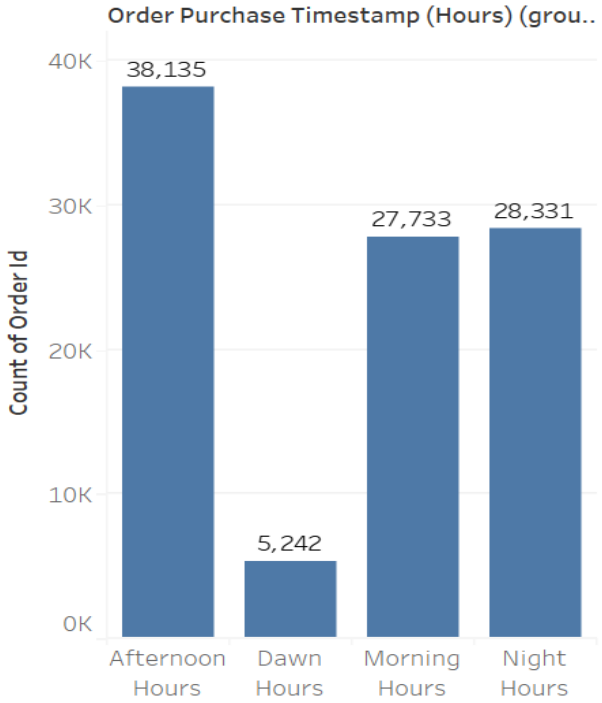


```
WITH CTE AS
(SELECT CASE
WHEN X.order_hour IN (0,1,2,3,4,5,6) THEN "Dawn"
WHEN X.order_hour IN (7,8,9,10,11,12) THEN "Morning"
WHEN X.order_hour IN (13,14,15,16,17,18) THEN "Afternoon"
WHEN X.order_hour IN (19,20,21,22,23) THEN "Night"
END AS OrderTime,
COUNT(DISTINCT order_id) Num_orders
FROM (SELECT *,
EXTRACT(HOUR FROM order_purchase_timestamp) order_hour
FROM [redacted].orders`) AS X
GROUP BY X.order_hour
ORDER BY X.order_hour DESC)

SELECT OrderTime,
SUM(Num_orders) TotalNumOrders
FROM CTE
GROUP BY OrderTime
ORDER BY TotalNumOrders DESC
```

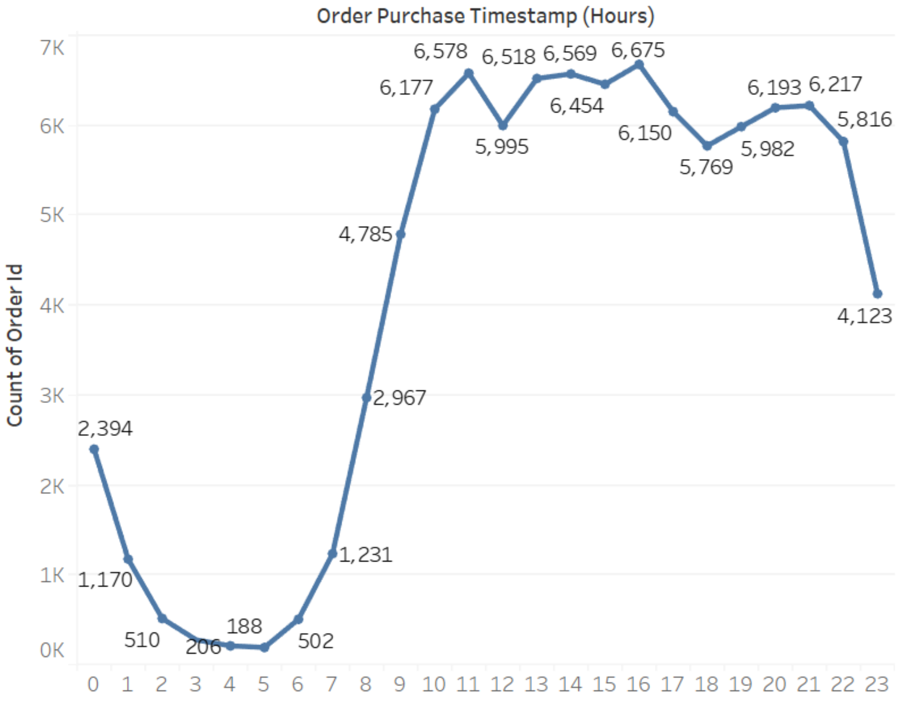
Query results

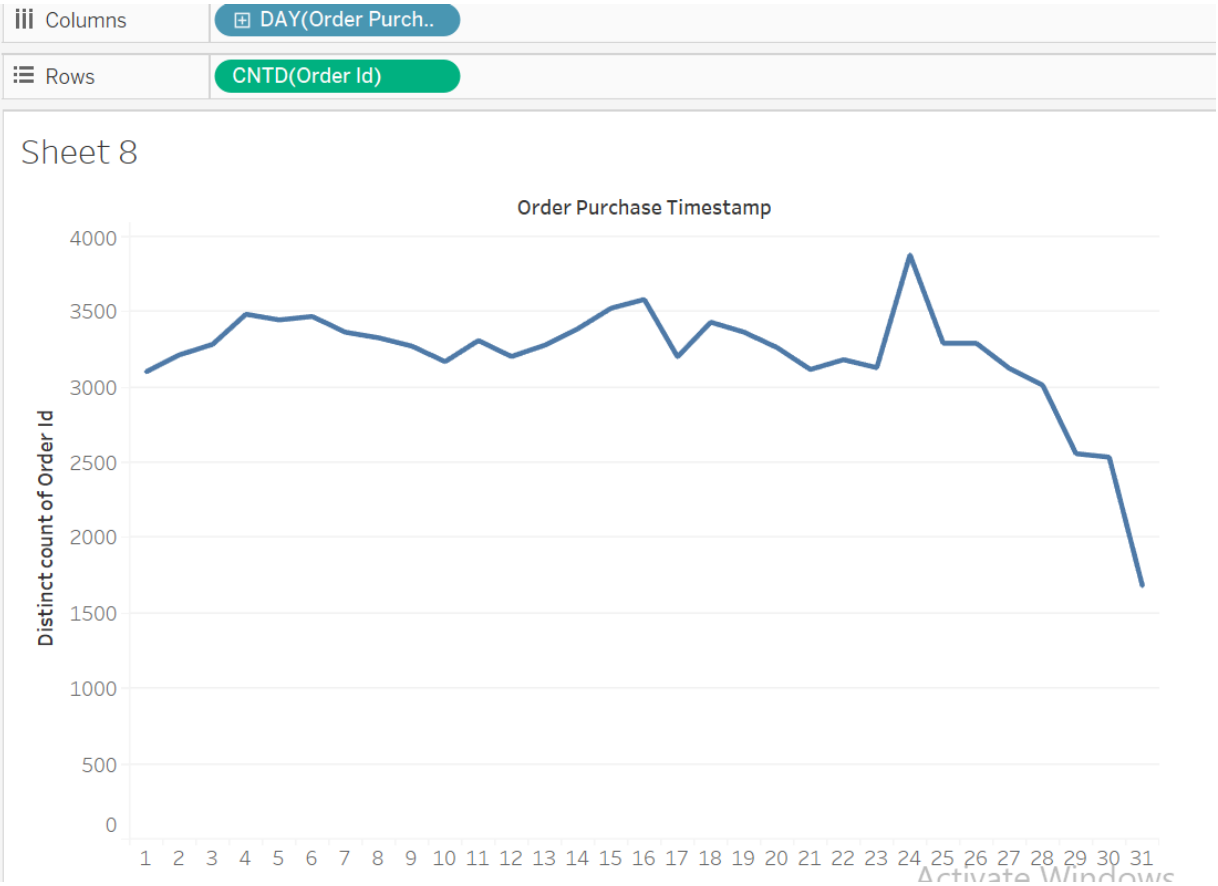
Row	OrderTime	TotalNumOrders
1	Afternoon	38135
2	Night	28331
3	Morning	27733
4	Dawn	5242

HourTrend



HourlyOrders





Insights:

- There is a growing trend in the number of orders from 2016 to 2018 with a significant drop in September 2018
- There is only last quarter data for 2016.
- There is no apparent monthly seasonality observed for the given period but in general there is a hike and drop every alternate month. (There's an increase in the number of orders in March, May, August as compared to previous months)
- Brazilian customers mostly place their order during Afternoon, followed by Night, Morning, Dawn.
- Hours when Brazilian customers have mostly ordered are between 10 to 17 hours and 19 to 22 hours.

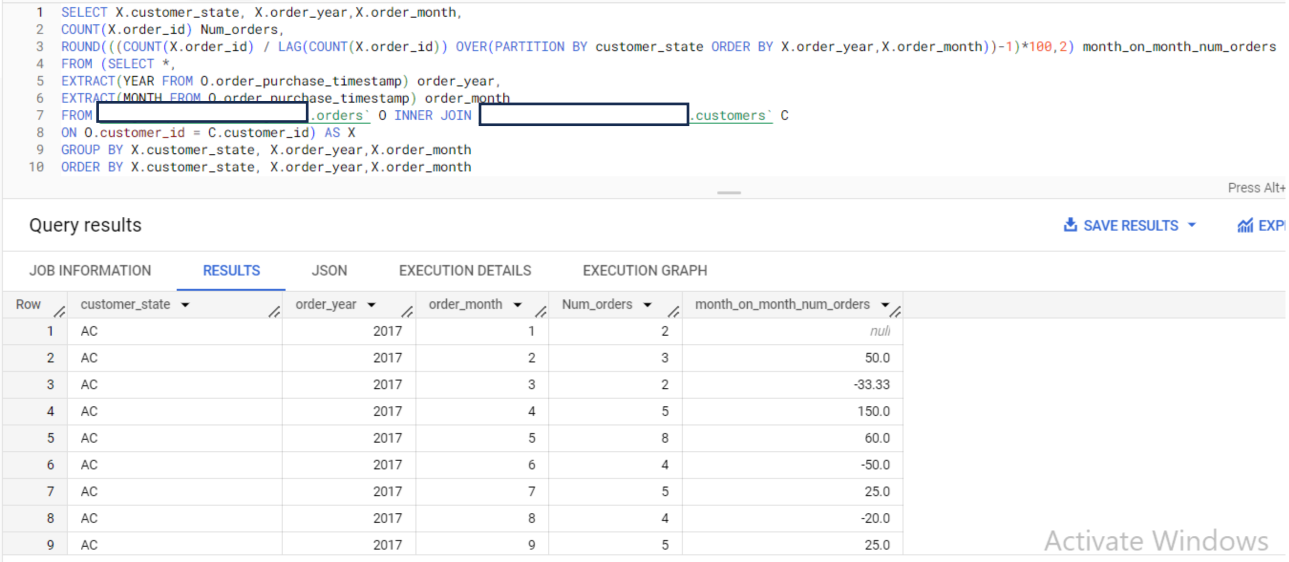
Recommendations:

- It would be worth checking the reason for drop in the count of orders in September 2018.
- Having more data would certainly help in understanding the seasonality factor better for eg., if users tend to shop more a month before or after their bonus month or if users tend to shop closer to their salary day each month.
- Display product recommendations / advertisements to users based on their search history during Afternoon hours when users tend to shop the most i.e. between 10 to 17 and 19 to 22 hours. This will result in users being lured into shopping what they were looking for.

- The number of orders tend to decrease closer to salary date (end of month) and increases on 1st until 24th /25th of each month hence product recommendations / advertisements should be mainly focused on these days to help increase sales.

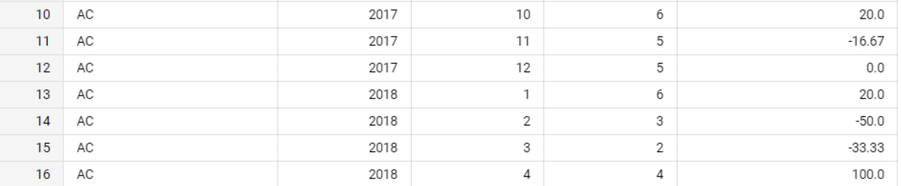
Evolution of E-commerce orders in the Brazil region:

➢ Get the month-on-month no. of orders placed in each state

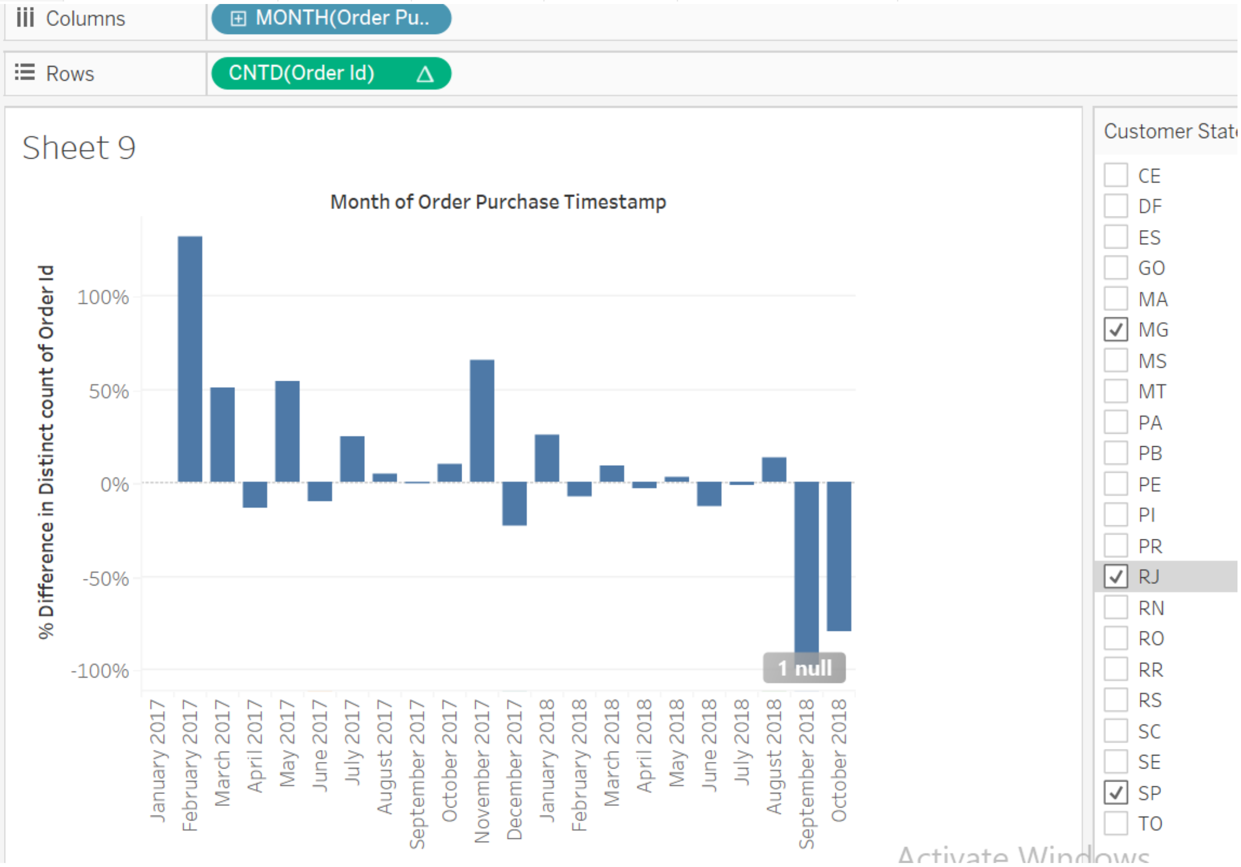


```
SELECT X.customer_state, X.order_year,X.order_month,
COUNT(X.order_id) Num_orders,
ROUND(((COUNT(X.order_id) / LAG(COUNT(X.order_id)) OVER(PARTITION BY customer_state ORDER BY X.order_year,X.order_month))-1)*100,2) month_on_month_num_orders
FROM (SELECT *,
EXTRACT(YEAR FROM O.order_purchase_timestamp) order_year,
EXTRACT(MONTH FROM O.order_purchase_timestamp) order_month
FROM [redacted].orders` O INNER JOIN [redacted].customers` C
ON O.customer_id = C.customer_id) AS X
GROUP BY X.customer_state, X.order_year,X.order_month
ORDER BY X.customer_state, X.order_year,X.order_month
```

Row	customer_state	order_year	order_month	Num_orders	month_on_month_num_orders
1	AC	2017	1	2	null
2	AC	2017	2	3	50.0
3	AC	2017	3	2	-33.33
4	AC	2017	4	5	150.0
5	AC	2017	5	8	60.0
6	AC	2017	6	4	-50.0
7	AC	2017	7	5	25.0
8	AC	2017	8	4	-20.0
9	AC	2017	9	5	25.0



10	AC	2017	10	6	20.0
11	AC	2017	11	5	-16.67
12	AC	2017	12	5	0.0
13	AC	2018	1	6	20.0
14	AC	2018	2	3	-50.0
15	AC	2018	3	2	-33.33
16	AC	2018	4	4	100.0



Columns: MONTH(Order Pu..

Rows: CNTD(Order Id) △

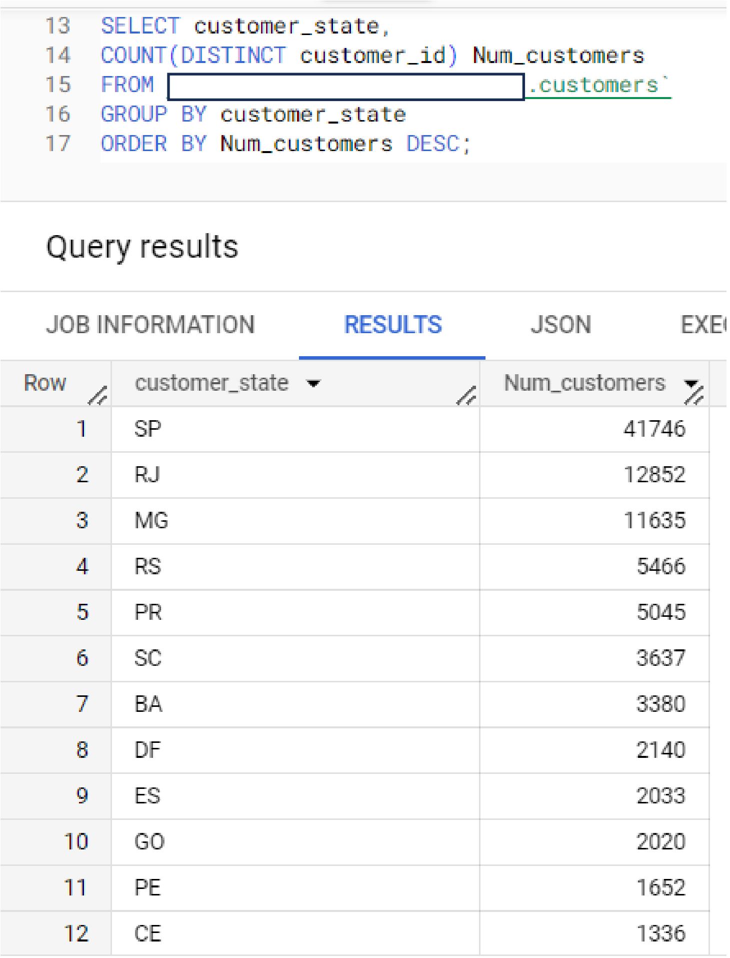
Sheet 9

Customer Stat

- □ CE
- □ DF
- □ ES
- □ GO
- □ MA
- ☑ MG
- □ MS
- □ MT
- □ PA
- □ PB
- □ PE
- □ PI
- □ PR
- ☑ RJ
- □ RN
- □ RO
- □ RR
- □ RS
- □ SC
- □ SE
- ☑ SP
- □ TO

Activate Windows

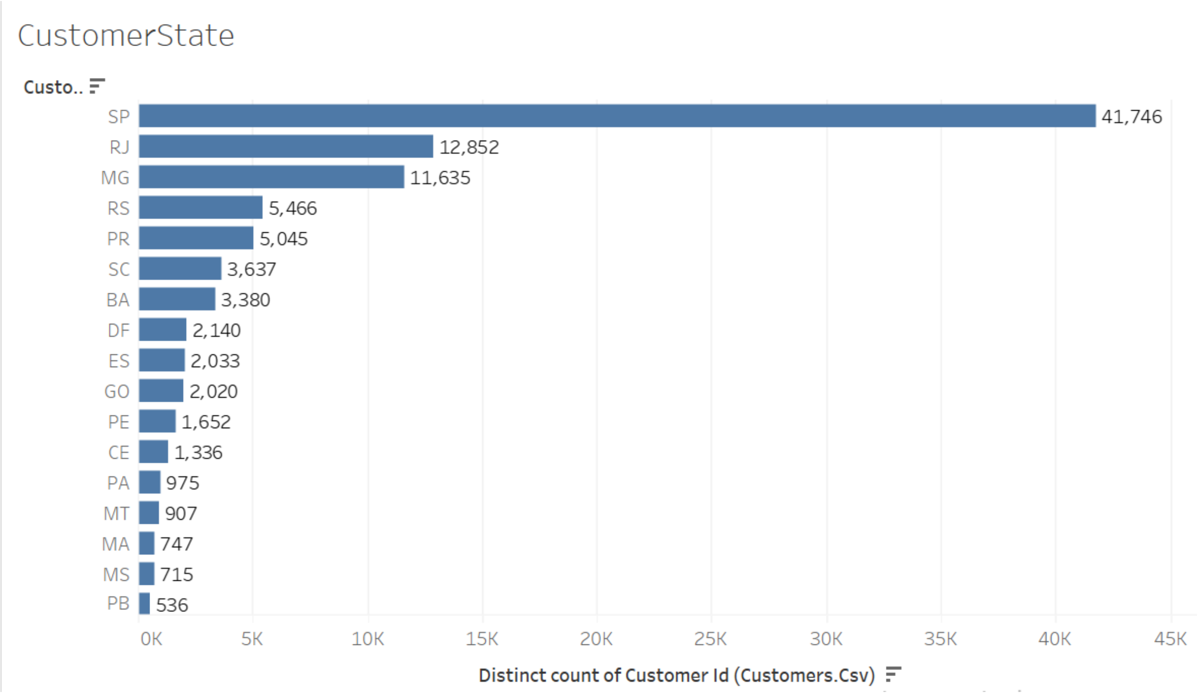
- How are the customers distributed across all the states?



```
SELECT customer_state,
COUNT(DISTINCT customer_id) Num_customers
FROM [redacted].customers`
GROUP BY customer_state
ORDER BY Num_customers DESC;
```

Query results

Row	customer_state	Num_customers
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
11	PE	1652
12	CE	1336



Insights:

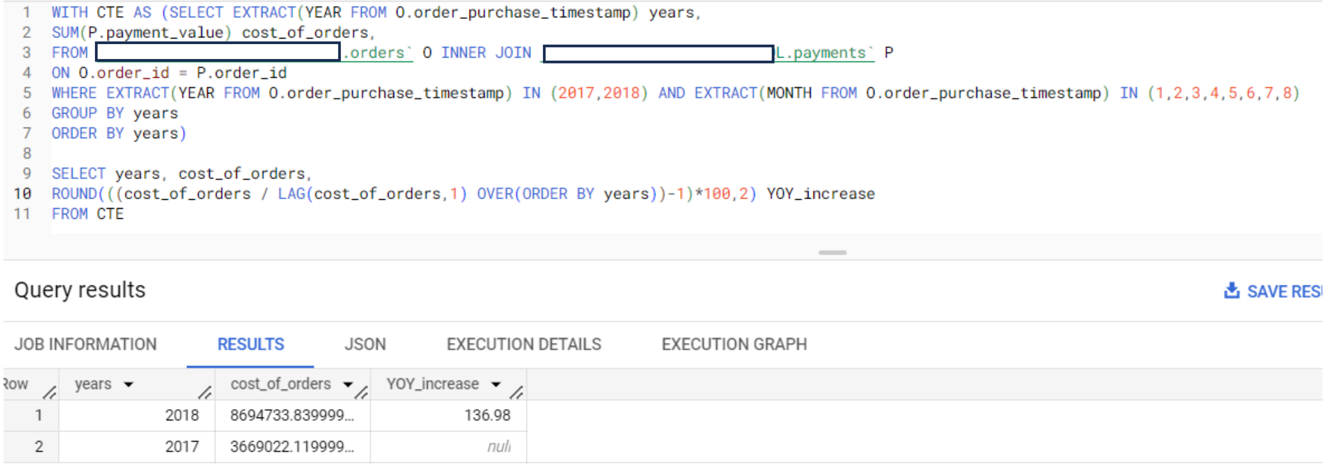
- The month-on-month number of orders percentage of Top 3 states with the greatest number of customers (SP, RJ, MG) shows a declining trend.
- Over 66.60% (66233 / 99441) Brazilian customers are from SP, RJ, MG states alone.
- Hence, assuming most of the sales come from customers ordering from these states.

Recommendations:

- Majority customer concentration is from only 3 out of 27 states, hence the recommendation is to have a strong foothold and increase marketing in other potentially high sales states like RS, PR, SC, BA as well to embody some diversification.
- Since the month-on-month number of orders for SP, RJ, MG states shows a declining trend, offer attractive products to increase the number of orders in these top states.
- Whilst also ensuring that the loyal customers stay loyal in states SP, RJ, MG.

Impact on Economy: Money movement by e-commerce by looking at order prices, freight and others.

- Get the % increase in the cost of orders from year 2017 to 2018 (include months between Jan to Aug only).

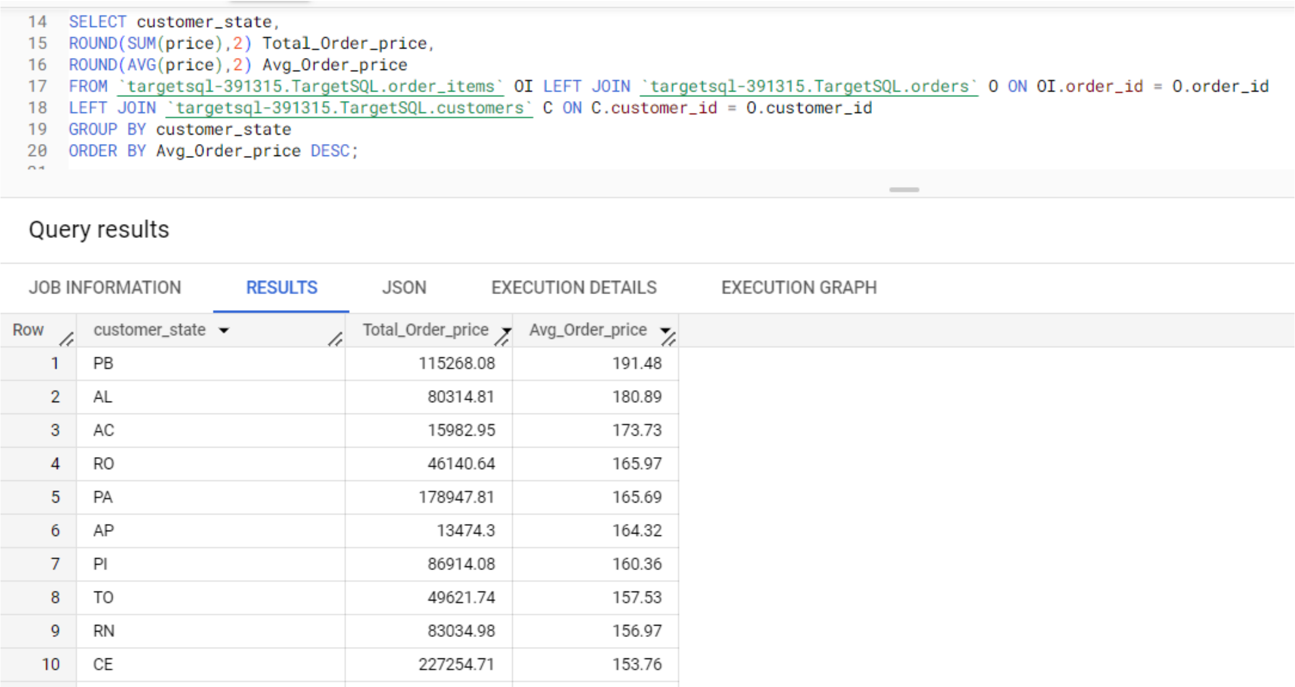


```
WITH CTE AS (SELECT EXTRACT(YEAR FROM O.order_purchase_timestamp) years,
SUM(P.payment_value) cost_of_orders,
FROM [redacted].orders` O INNER JOIN [redacted]L.payments` P
ON O.order_id = P.order_id
WHERE EXTRACT(YEAR FROM O.order_purchase_timestamp) IN (2017,2018) AND EXTRACT(MONTH FROM O.order_purchase_timestamp) IN (1,2,3,4,5,6,7,8)
GROUP BY years
ORDER BY years)

SELECT years, cost_of_orders,
ROUND(((cost_of_orders / LAG(cost_of_orders,1) OVER(ORDER BY years))-1)*100,2) YOY_increase
FROM CTE
```

Query results

Row	years	cost_of_orders	YOY_increase
1	2018	8694733.839999…	136.98
2	2017	3669022.119999…	null

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

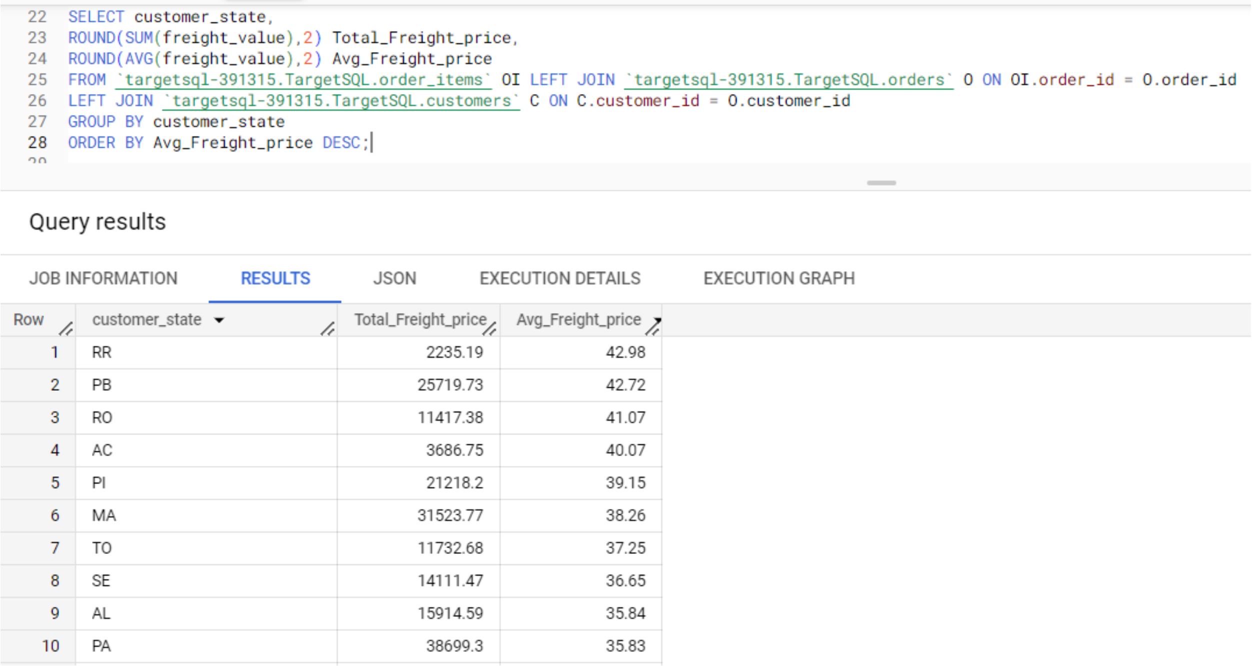


```
SELECT customer_state,
ROUND(SUM(price),2) Total_Order_price,
ROUND(AVG(price),2) Avg_Order_price
FROM `targetsql-391315.TargetSQL.order_items` OI LEFT JOIN `targetsql-391315.TargetSQL.orders` O ON OI.order_id = O.order_id
LEFT JOIN `targetsql-391315.TargetSQL.customers` C ON C.customer_id = O.customer_id
GROUP BY customer_state
ORDER BY Avg_Order_price DESC;
```

Query results

Row	customer_state	Total_Order_price	Avg_Order_price
1	PB	115268.08	191.48
2	AL	80314.81	180.89
3	AC	15982.95	173.73
4	RO	46140.64	165.97
5	PA	178947.81	165.69
6	AP	13474.3	164.32
7	PI	86914.08	160.36
8	TO	49621.74	157.53
9	RN	83034.98	156.97
10	CE	227254.71	153.76

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



```sql
SELECT customer_state,
ROUND(SUM(freight_value),2) Total_Freight_price,
ROUND(AVG(freight_value),2) Avg_Freight_price
FROM `targetsql-391315.TargetSQL.order_items` OI LEFT JOIN `targetsql-391315.TargetSQL.orders` O ON OI.order_id = O.order_id
LEFT JOIN `targetsql-391315.TargetSQL.customers` C ON C.customer_id = O.customer_id
GROUP BY customer_state
ORDER BY Avg_Freight_price DESC;
```

Query results

Row	customer_state	Total_Freight_price	Avg_Freight_price
1	RR	2235.19	42.98
2	PB	25719.73	42.72
3	RO	11417.38	41.07
4	AC	3686.75	40.07
5	PI	21218.2	39.15
6	MA	31523.77	38.26
7	TO	11732.68	37.25
8	SE	14111.47	36.65
9	AL	15914.59	35.84
10	PA	38699.3	35.83

<u>Insights:</u>

- There is almost 137% increase in the cost of orders in 2018 as opposed to 2017
- PB, AL, AC, RO, PA, PI, TO are in the Top 10 states with highest average order price and highest average freight price.
- This could be one of the major reasons for declining month-on-month number of orders for Top 3 cities with the greatest number of customers who have ordered from XYZ Co..

<u>Recommendation:</u>

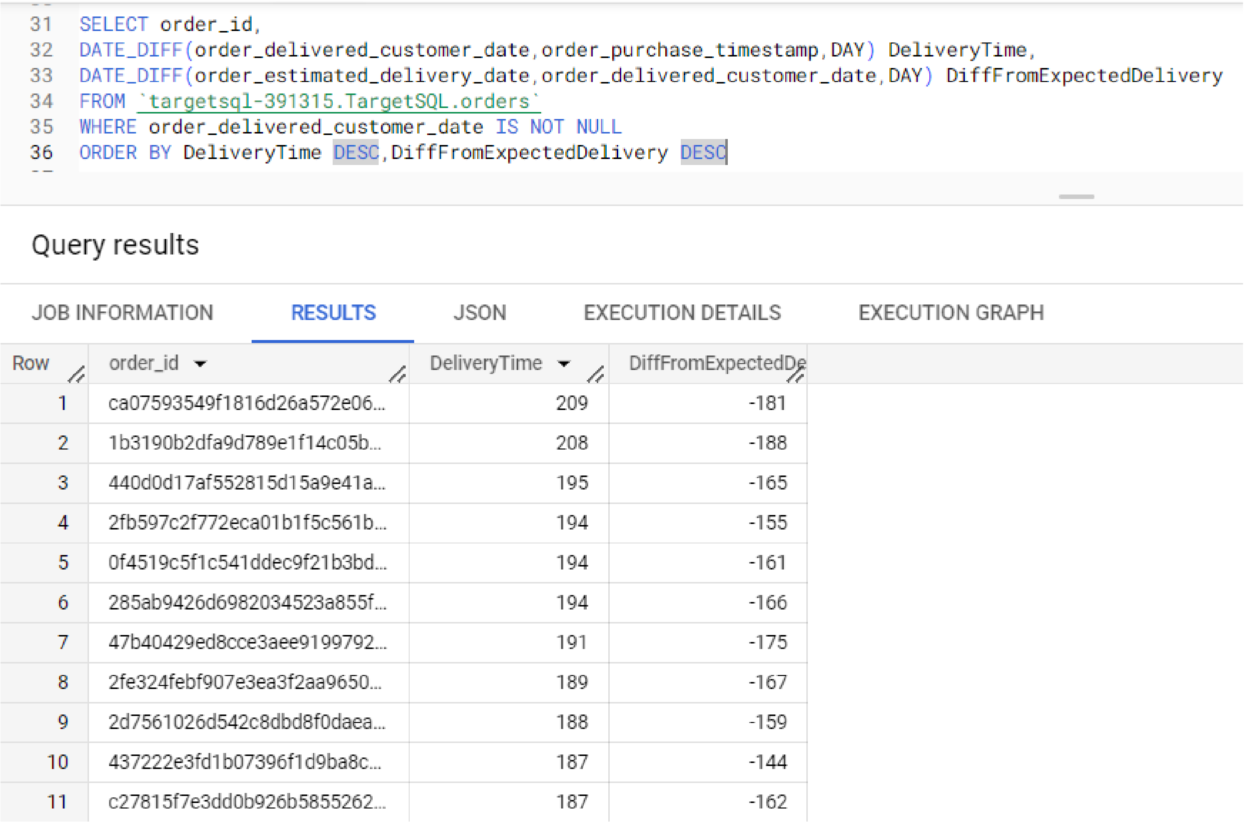
- Look to reduce the freight price, thus the cost of orders to generate more profit.
  - Either the cost can be passed onto the consumers (which probably is the case here) - but this will result in declining sales or orders.
  - Or absorb the cost so sales do not decline but this will result in reduced profit hence impact stakeholders.

Analysis based on sales, freight and delivery time.

- ➢ 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:

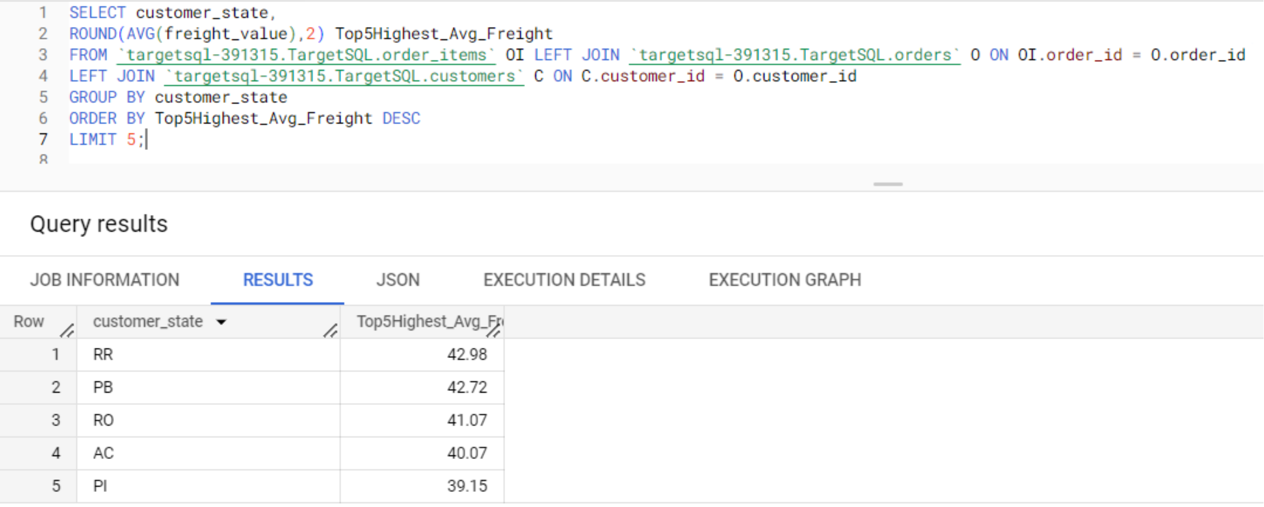
- **time_to_deliver** = order_delivered_customer_date - order_purchase_timestamp
- **diff_estimated_delivery** = order_estimated_delivery_date - order_delivered_customer_date



```
SELECT order_id,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) DeliveryTime,
DATE_DIFF(order_estimated_delivery_date,order_delivered_customer_date,DAY) DiffFromExpectedDelivery
FROM `targetsql-391315.TargetSQL.orders`
WHERE order_delivered_customer_date IS NOT NULL
ORDER BY DeliveryTime DESC,DiffFromExpectedDelivery DESC
```

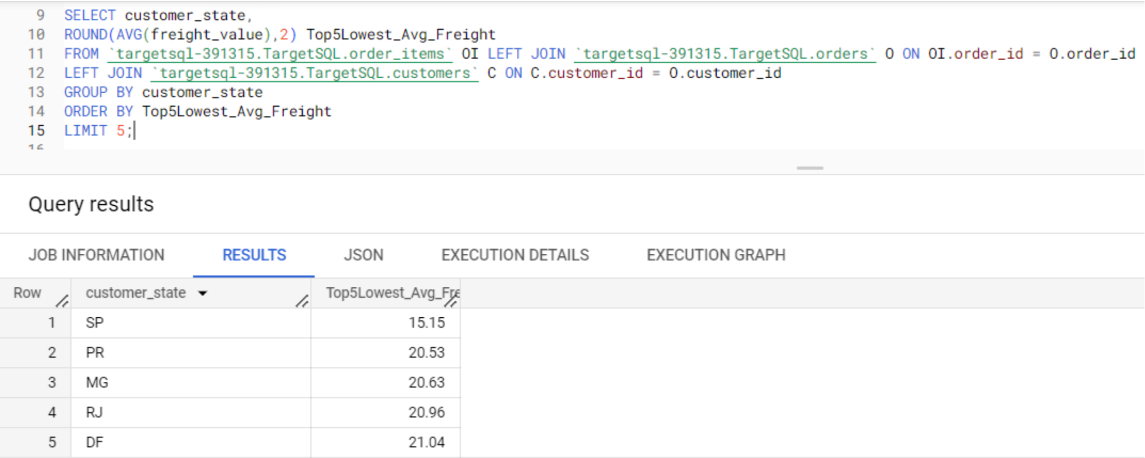
Row	order_id	DeliveryTime	DiffFromExpectedDe
1	ca07593549f1816d26a572e06...	209	-181
2	1b3190b2dfa9d789e1f14c05b...	208	-188
3	440d0d17af552815d15a9e41a...	195	-165
4	2fb597c2f772eca01b1f5c561b...	194	-155
5	0f4519c5f1c541ddec9f21b3bd...	194	-161
6	285ab9426d6982034523a855f...	194	-166
7	47b40429ed8cce3aee9199792...	191	-175
8	2fe324febf907e3ea3f2aa9650...	189	-167
9	2d7561026d542c8dbd8f0daea...	188	-159
10	437222e3fd1b07396f1d9ba8c...	187	-144
11	c27815f7e3dd0b926b5855262...	187	-162

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



```
SELECT customer_state,
ROUND(AVG(freight_value),2) Top5Highest_Avg_Freight
FROM `targetsql-391315.TargetSQL.order_items` OI LEFT JOIN `targetsql-391315.TargetSQL.orders` O ON OI.order_id = O.order_id
LEFT JOIN `targetsql-391315.TargetSQL.customers` C ON C.customer_id = O.customer_id
GROUP BY customer_state
ORDER BY Top5Highest_Avg_Freight DESC
LIMIT 5;
```

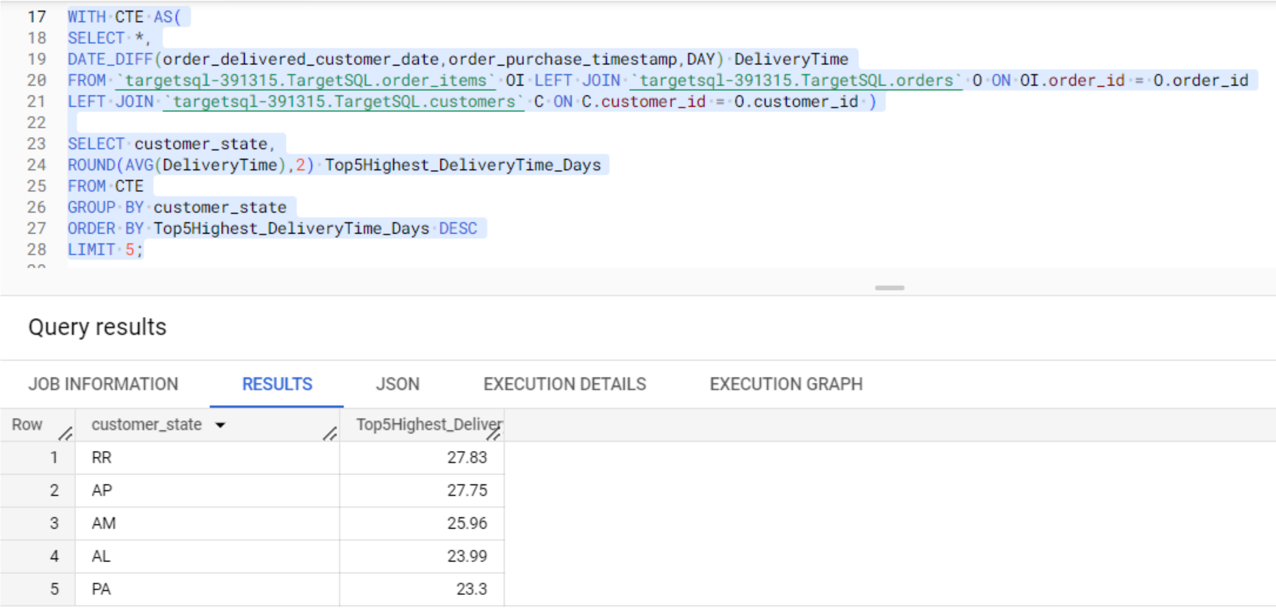
Row	customer_state	Top5Highest_Avg_Fr
1	RR	42.98
2	PB	42.72
3	RO	41.07
4	AC	40.07
5	PI	39.15



```
SELECT customer_state,
ROUND(AVG(freight_value),2) Top5Lowest_Avg_Freight
FROM `targetsql-391315.TargetSQL.order_items` OI LEFT JOIN `targetsql-391315.TargetSQL.orders` O ON OI.order_id = O.order_id
LEFT JOIN `targetsql-391315.TargetSQL.customers` C ON C.customer_id = O.customer_id
GROUP BY customer_state
ORDER BY Top5Lowest_Avg_Freight
LIMIT 5;
```

Row	customer_state	Top5Lowest_Avg_Fre
1	SP	15.15
2	PR	20.53
3	MG	20.63
4	RJ	20.96
5	DF	21.04

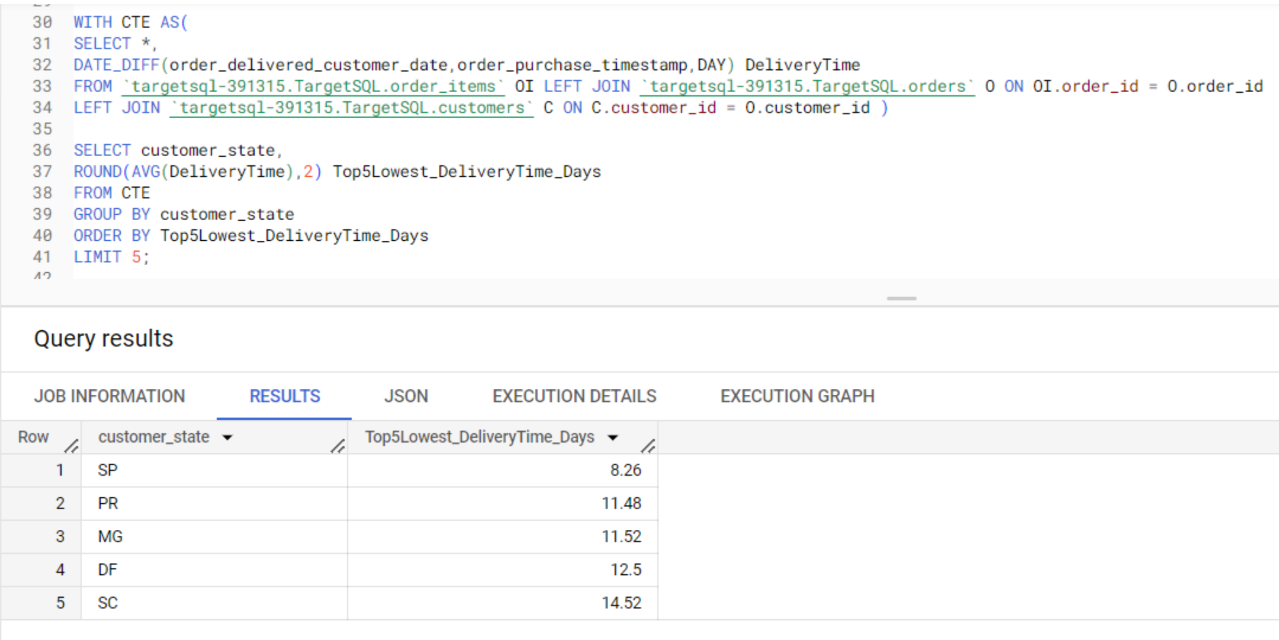
- Find out the top 5 states with the highest & lowest average delivery time.



```sql
WITH CTE AS(
SELECT *,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) DeliveryTime
FROM `targetsql-391315.TargetSQL.order_items` OI LEFT JOIN `targetsql-391315.TargetSQL.orders` O ON OI.order_id = O.order_id
LEFT JOIN `targetsql-391315.TargetSQL.customers` C ON C.customer_id = O.customer_id )

SELECT customer_state,
ROUND(AVG(DeliveryTime),2) Top5Highest_DeliveryTime_Days
FROM CTE
GROUP BY customer_state
ORDER BY Top5Highest_DeliveryTime_Days DESC
LIMIT 5;
```

Query results

Row	customer_state	Top5Highest_Deliver
1	RR	27.83
2	AP	27.75
3	AM	25.96
4	AL	23.99
5	PA	23.3

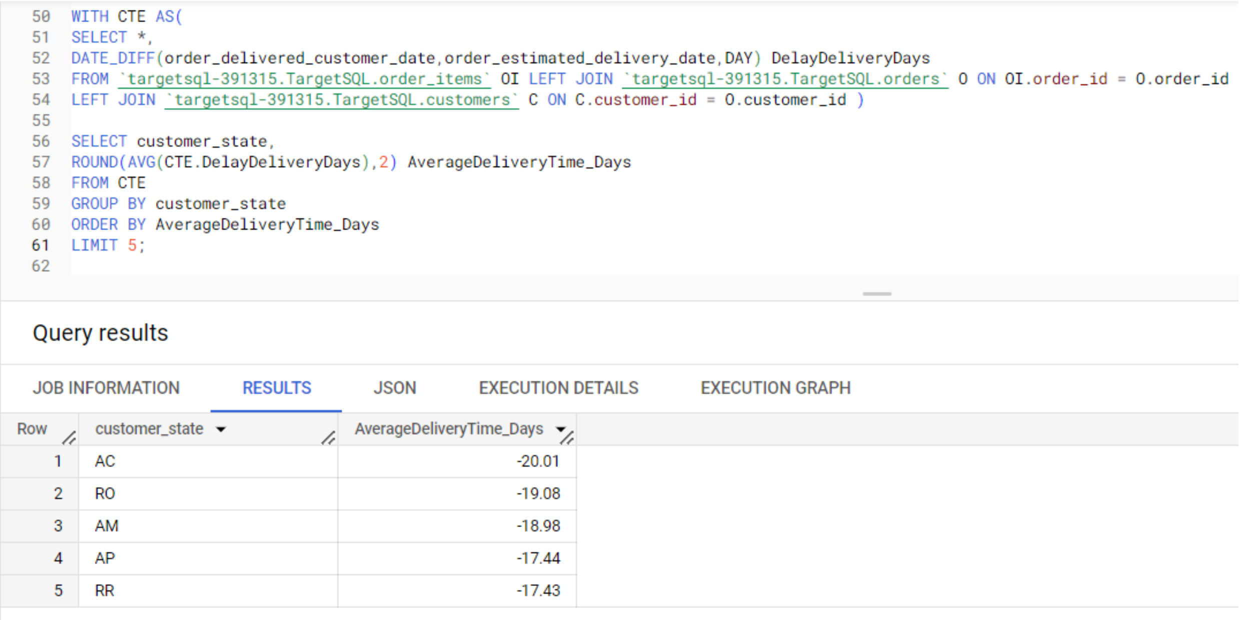


```sql
WITH CTE AS(
SELECT *,
DATE_DIFF(order_delivered_customer_date,order_purchase_timestamp,DAY) DeliveryTime
FROM `targetsql-391315.TargetSQL.order_items` OI LEFT JOIN `targetsql-391315.TargetSQL.orders` O ON OI.order_id = O.order_id
LEFT JOIN `targetsql-391315.TargetSQL.customers` C ON C.customer_id = O.customer_id )

SELECT customer_state,
ROUND(AVG(DeliveryTime),2) Top5Lowest_DeliveryTime_Days
FROM CTE
GROUP BY customer_state
ORDER BY Top5Lowest_DeliveryTime_Days
LIMIT 5;
```

Query results

Row	customer_state	Top5Lowest_DeliveryTime_Days
1	SP	8.26
2	PR	11.48
3	MG	11.52
4	DF	12.5
5	SC	14.52

- 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.



```
WITH CTE AS(
SELECT *,
DATE_DIFF(order_delivered_customer_date,order_estimated_delivery_date,DAY) DelayDeliveryDays
FROM `targetsql-391315.TargetSQL.order_items` OI LEFT JOIN `targetsql-391315.TargetSQL.orders` O ON OI.order_id = O.order_id
LEFT JOIN `targetsql-391315.TargetSQL.customers` C ON C.customer_id = O.customer_id )

SELECT customer_state,
ROUND(AVG(CTE.DelayDeliveryDays),2) AverageDeliveryTime_Days
FROM CTE
GROUP BY customer_state
ORDER BY AverageDeliveryTime_Days
LIMIT 5;
```

Query results

Row	customer_state	AverageDeliveryTime_Days
1	AC	-20.01
2	RO	-19.08
3	AM	-18.98
4	AP	-17.44
5	RR	-17.43

Insights:

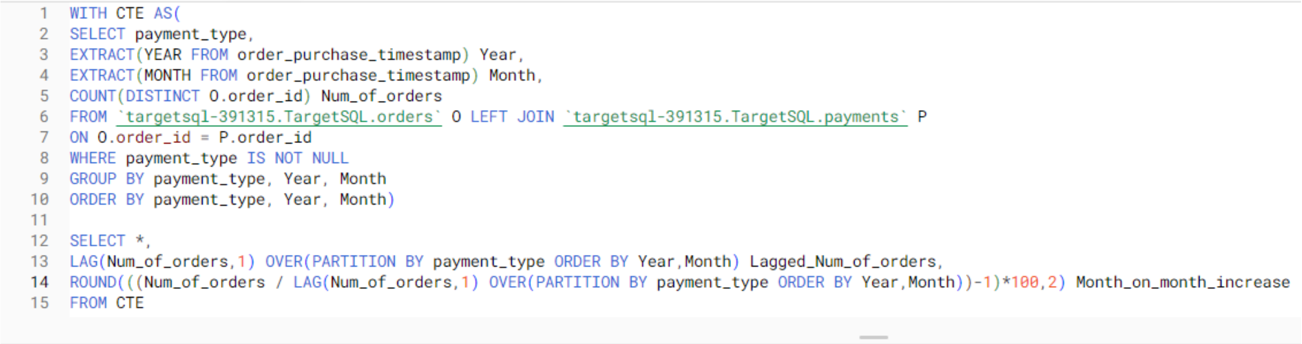
- Delivery for some orders is massively delayed by ~181 days (almost half a year)
- SP and MG, states with the greatest number of customers, also have the lowest average freight price, and delivery time.
- RR, AP, AM, AL, PA are states with highest average delivery time of ~23 to 27 days
- States AC, RO, AM, AP, RR have average delivery time ~17 to 20 days before estimated delivery dates.
- PB, AL, AC, RO, PA, PI, TO are in the Top 10 states with the highest average order price and highest average freight price.

Recommendation:

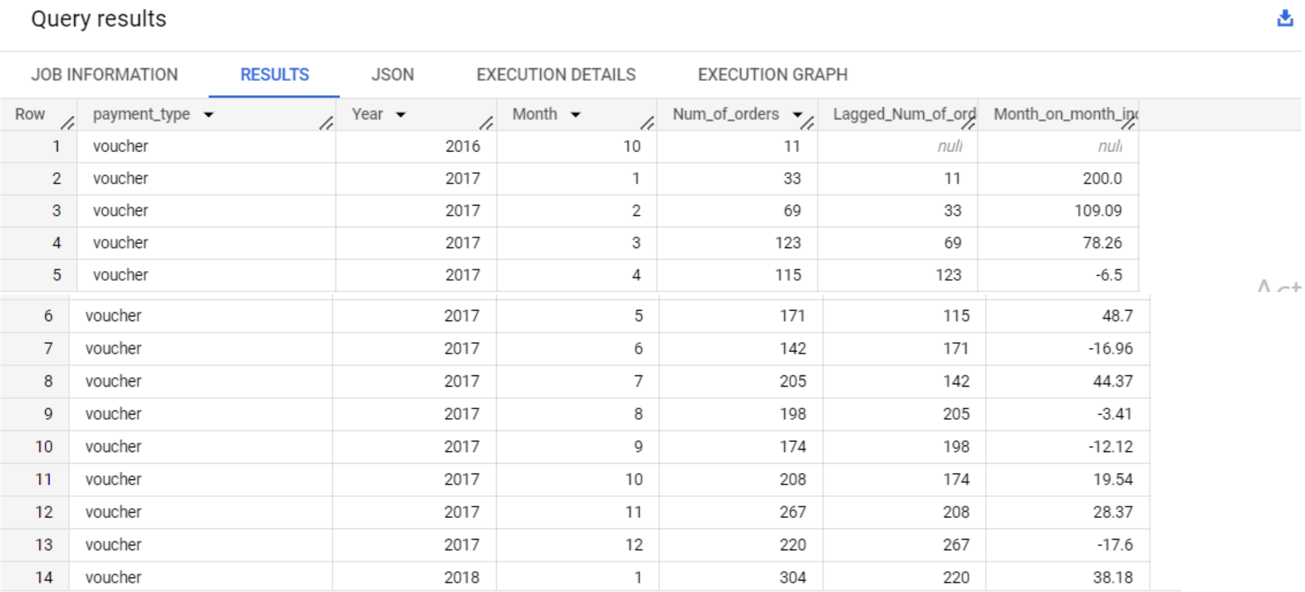
- Delivery time of orders needs to be reduced which will attract more customers.

Analysis based on the payments:

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

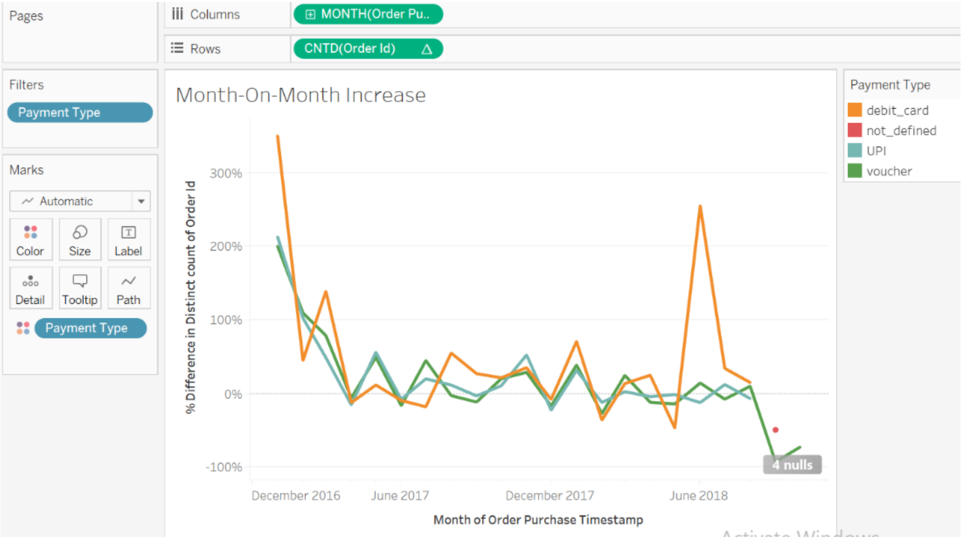


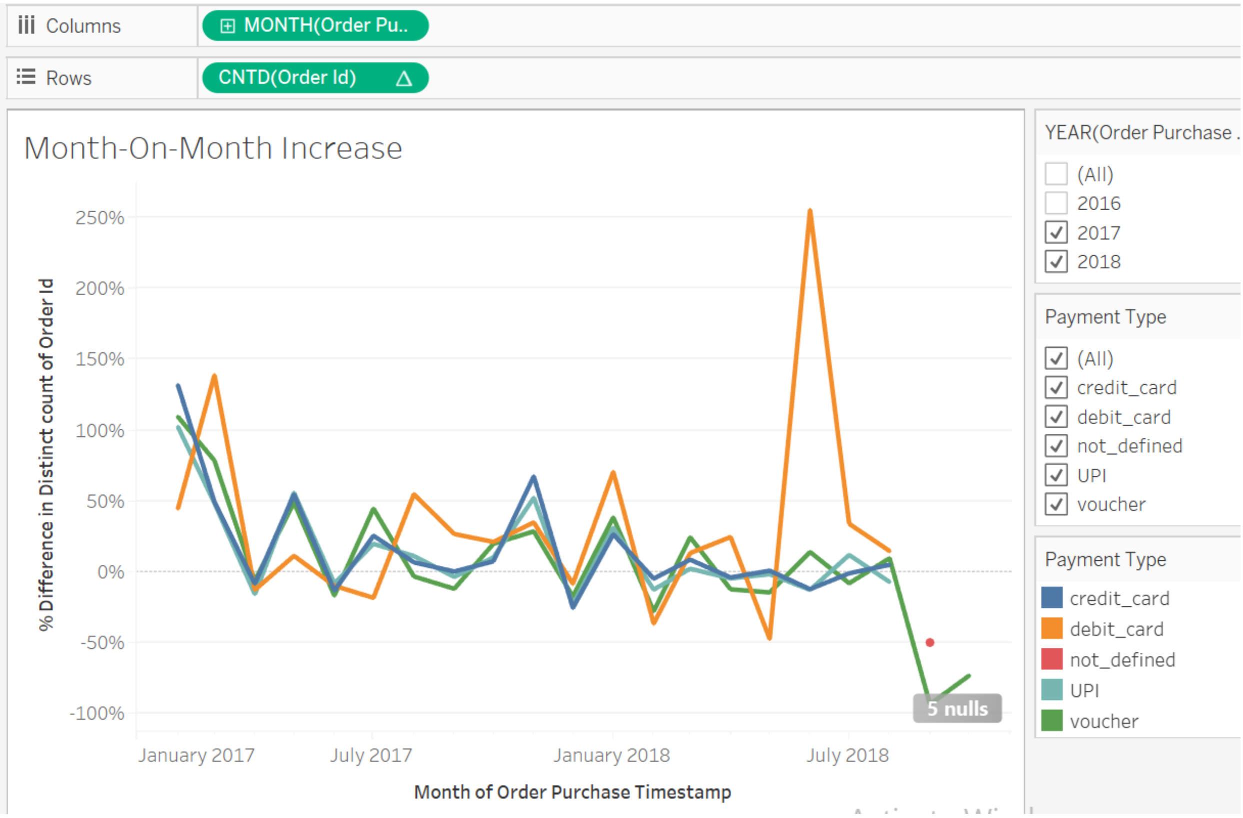
```sql
WITH CTE AS(
SELECT payment_type,
EXTRACT(YEAR FROM order_purchase_timestamp) Year,
EXTRACT(MONTH FROM order_purchase_timestamp) Month,
COUNT(DISTINCT O.order_id) Num_of_orders
FROM `targetsql-391315.TargetSQL.orders` O LEFT JOIN `targetsql-391315.TargetSQL.payments` P
ON O.order_id = P.order_id
WHERE payment_type IS NOT NULL
GROUP BY payment_type, Year, Month
ORDER BY payment_type, Year, Month)

SELECT *,
LAG(Num_of_orders,1) OVER(PARTITION BY payment_type ORDER BY Year,Month) Lagged_Num_of_orders,
ROUND(((Num_of_orders / LAG(Num_of_orders,1) OVER(PARTITION BY payment_type ORDER BY Year,Month))-1)*100,2) Month_on_month_increase
FROM CTE
```



Query results

Row	payment_type	Year	Month	Num_of_orders	Lagged_Num_of_ord	Month_on_month_in
1	voucher	2016	10	11	null	null
2	voucher	2017	1	33	11	200.0
3	voucher	2017	2	69	33	109.09
4	voucher	2017	3	123	69	78.26
5	voucher	2017	4	115	123	-6.5
6	voucher	2017	5	171	115	48.7
7	voucher	2017	6	142	171	-16.96
8	voucher	2017	7	205	142	44.37
9	voucher	2017	8	198	205	-3.41
10	voucher	2017	9	174	198	-12.12
11	voucher	2017	10	208	174	19.54
12	voucher	2017	11	267	208	28.37
13	voucher	2017	12	220	267	-17.6
14	voucher	2018	1	304	220	38.18





Columns: MONTH(Order Pu..

Rows: CNTD(Order Id) △

YEAR(Order Purchase .

- ☐ (All)
- ☐ 2016
- ☑ 2017
- ☑ 2018

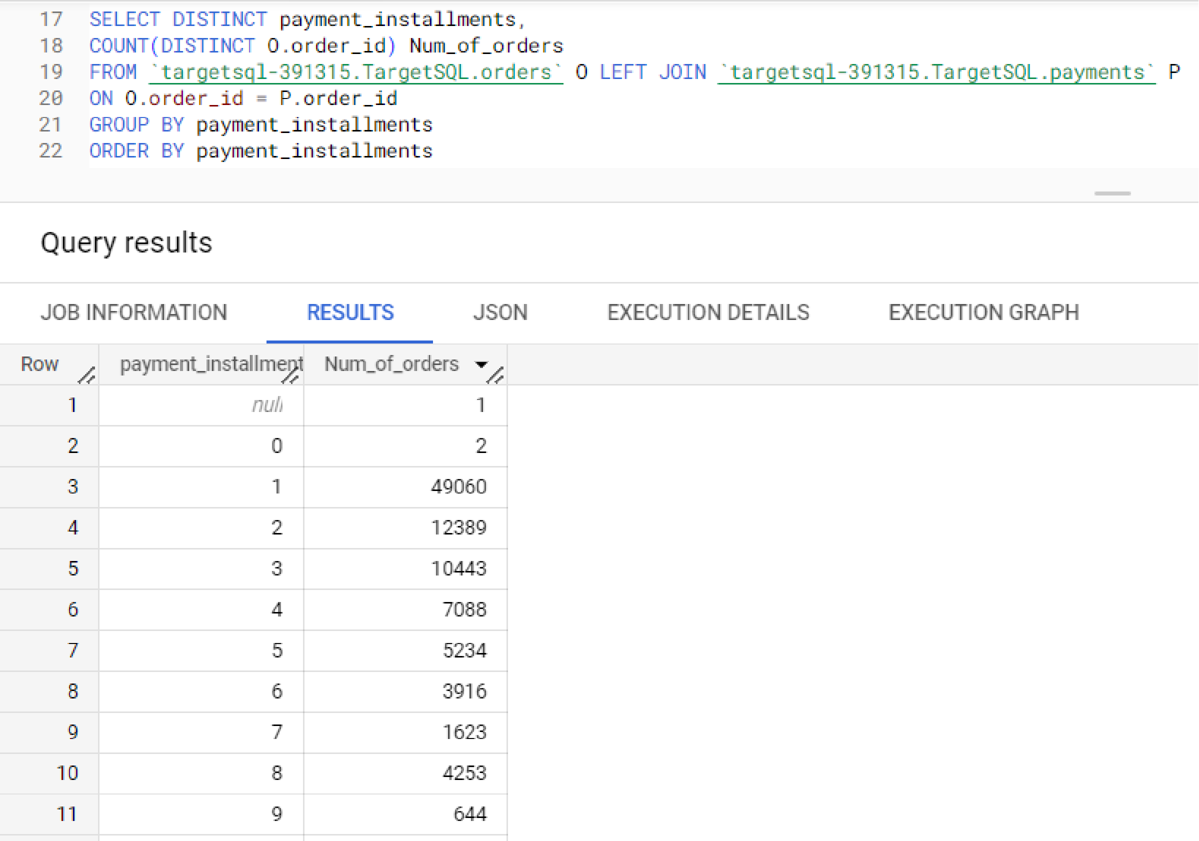
Payment Type

- ☑ (All)
- ☑ credit_card
- ☑ debit_card
- ☑ not_defined
- ☑ UPI
- ☑ voucher

Payment Type

- 🟦 credit_card
- 🟧 debit_card
- 🟥 not_defined
- 🟦 UPI
- 🟩 voucher

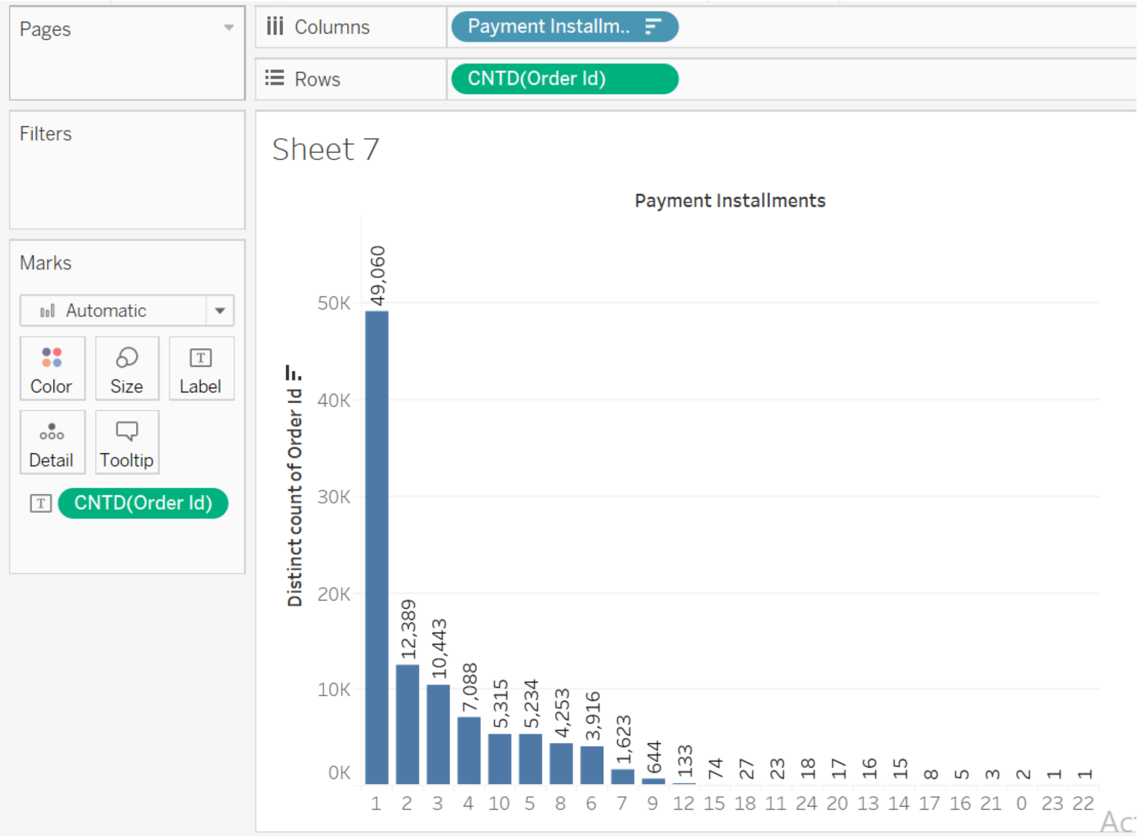
➢ Find the no. of orders placed on the basis of the payment installments that have been paid.



```
SELECT DISTINCT payment_installments,
COUNT(DISTINCT O.order_id) Num_of_orders
FROM `targetsql-391315.TargetSQL.orders` O LEFT JOIN `targetsql-391315.TargetSQL.payments` P
ON O.order_id = P.order_id
GROUP BY payment_installments
ORDER BY payment_installments
```

Query results

Row	payment_installment	Num_of_orders
1	null	1
2	0	2
3	1	49060
4	2	12389
5	3	10443
6	4	7088
7	5	5234
8	6	3916
9	7	1623
10	8	4253
11	9	644



Insights:

- Since 2016 only has data for one month, skipping that to calculate month-on-month number of orders.
- Overall, there is a percentage decline in the month-on-month number of orders placed using each payment method – another indicator of reduction in total sales over time.
- There is still some stability in payments via debit card but other options are gradually declining.
- Maximum number of orders have payment installments ranging from 1 to 10, with 1 being the most preferred, followed by 2,3,4,10

Recommendation:

- Steps to be taken to ensure the percentage decline in number of orders is catered to as it is seen as an indicator though a lot of factors.
- Credit card, UPI, voucher are preferably not seen as more attractive as opposed to debit cards hence make tie-ups with banks to include offers for debit card to purchase XYZ Co. products.
- Provide attractive offers for 1 installment payment or payments with 2,3,4,10 installments as they seem to be the most preferred choice amongst Brazilian demographics.

Overall Actionable Insights & Recommendations

Actionable Insights:

- The dataset available is of customers' orders placed from 2016 to 2018 from over 27 different states & 4119 different cities in Brazil operations of XYZ Co. company.
- Note: Since, 2016 has only one month of data for last quarter, skipping that to calculate month-on-month data.
- Customer Demographics Data:
  - Total customers: 99,441
  - Over 15.62% (15540 / 99441) customers of XYZ Co. in Brazil are from Sao Paulo city.
  - Over 35.23% (35042 / 99441) customers are in the Top 10 cities alone.
  - Over 66.60% (66233 / 99441) Brazilian customers are from SP, RJ, MG states alone.

- Growing trend in the number of orders from 2016 to 2018 with a significant drop in Sep 2018.
- No monthly seasonality observed but in general there is a hike and drop every alternate month.
- Orders count decreases closer to salary date & increases from 1st until 24th /25th each month. (Assuming 28th – 31st of each month is salary day)
- Orders mostly placed during Afternoon, followed by Night, Morning, Dawn. (Most orders are placed between 10 to 17 hours and 19 to 22 hours.)

- Overall, there is a percentage decline in the month-on-month number of orders for Top 3 states with the greatest number of customers (SP, RJ, MG) and for each payment method.
- There is almost 137% increase in the cost of orders from 2017 to 2018.
- PB, AL, AC, RO, PA, PI, TO are one of the Top 10 states with highest average order price and highest average freight price.
- SP and MG state with the greatest number of customers, also have the lowest average freight price, and delivery time.
- Delivery for some orders is massively delayed by ~181 days (almost half a year)
- RR, AP, AM, AL, PA are states with highest average delivery time of ~23 to 27 days
- AC, RO, AM, AP, RR have average deliveries ~17 to 20 days before estimated delivery dates.

- There is still some stability in payments via debit-card but other options are gradually declining.
- Maximum number of orders have payment installments ranging from 1 to 10, with 1 being the most preferred, followed by 2,3,4,10

Recommendation:

- Considering 66.60% of the Brazilian customers are from 3 states SP, RJ, MG – It should be worth checking how much do orders from these states contribute towards total sales and profit in Brazil.
- Similar analysis for 15.62% demographics in Sao Paulo city and 35.23% in top 10 cities.
- It would be worth checking the reason for drop in the count of orders in September 2018 and additional data would help in understanding the seasonality factor better.

General Recommendation:

- To help increase sales, display product recommendations / advertisements to users based on their search history during Afternoon hours (between 10 to 17 and 19 to 22 hours) and mainly focused on 1st to 25th days of the month as users tend to shop the most during this period.
- Majority customer concentration is from only 3 out of 27 states in Brazil, hence the recommendation is to have a strong foothold and increase marketing in other potentially high sales states like RS, PR, SC, BA as well to embody some diversification whilst also ensuring that the loyal customers stay loyal in states like SP, RJ, MG.
- Although there is a growing trend for number of orders from 2017 to 2018, there is a percentage decline in the month-on-month number of orders for Top 3 states with the greatest number of customers (SP, RJ, MG) and for each payment method.
- Thus, assuming most of the sales come from customers in Top 3 states, there appears to be an overall decline in sales over time.
- Percentage decline in orders need to be catered to as it is being indicated by multiple factors like declining orders for each payment type and each Top 3 states (by customer count).

Cost Recommendation:

- Look to reduce the freight price, thus the cost of orders to generate more profit. If high cost of products is due to economic factors,
- ○ Either the cost can be passed onto the consumers (which probably is the case here) - but this will result in declining sales or orders.
- ○ Or sellers can absorb the cost so sales do not decline but this will result in reduced profit hence impact stakeholders.
- Cost-benefit analysis of freight price and order price is required to understand where cost can be reduced or absorbed while ensuring that there I not much decline in the sales. Elasticity of products needs to be considered while performing this analysis.

Delivery Recommendation:

- Some states like AC, RO, AM, AP, RR have average deliveries way before estimated delivery date, hence some motivation from these states to be taken to improve logistics in states that have significantly delayed delivery timings.
- Delivery time of orders needs to be reduced as some orders are delayed by as much as half a year.
- Focusing on improving logistics and connectivity would be beneficial in reducing delivery time and attract more customers, this will in turn help in increasing the number of orders and sales.

Payment Recommendation:

- Credit card, UPI, voucher are preferably not seen as more attractive as opposed to debit cards hence make tie-ups with banks to include offers for debit card to purchase XYZ Co. products.
- Provide attractive offers for 1 installment payment or payments with 2,3,4,10 installments as they seem to be the most preferred choice amongst Brazilian demographics.