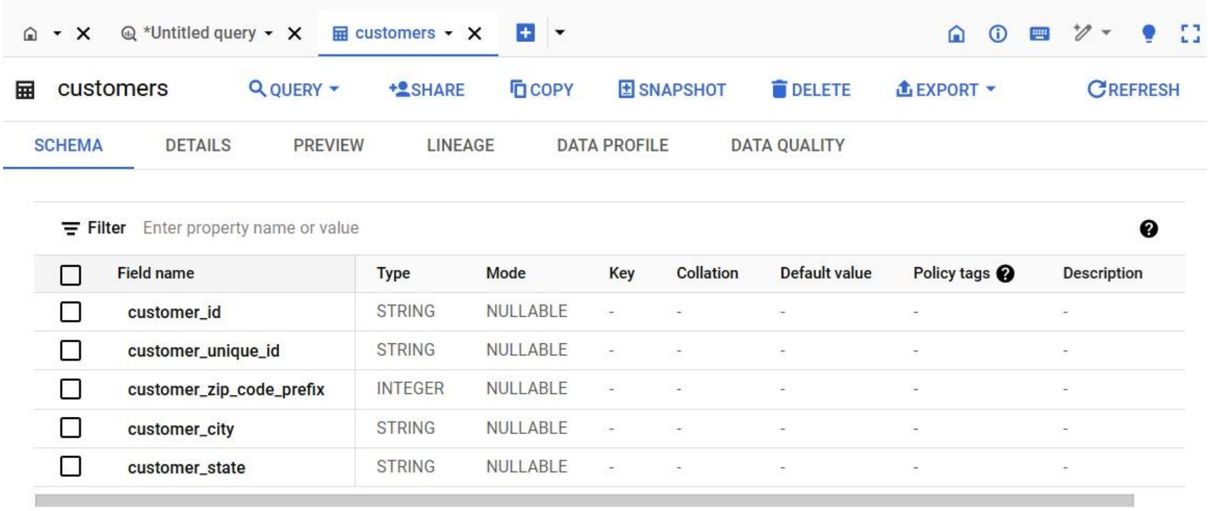
Business Case Study – TARGET SQL

Target is a globally renowned brand and a prominent retailer in the United States. Target makes itself a preferred shopping destination by offering outstanding value, inspiration, innovation and an exceptional guest experience that no other retailer can deliver.
This particular business case focuses on the operations of Target in Brazil and provides insightful information about 100,000 orders placed between 2016 and 2018. The dataset offers a comprehensive view of various dimensions including the order status, price, payment and freight performance, customer location, product attributes, and customer reviews.
By analysing this extensive dataset, it becomes possible to gain valuable insights into Target's operations in Brazil. The information can shed light on various aspects of the business, such as order processing, pricing strategies, payment and shipping efficiency, customer demographics, product characteristics, and customer satisfaction levels.

Q1: Import the dataset and do usual exploratory analysis steps like checking the structure & characteristics of the dataset:

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

Answer:



	Field name	Type	Mode	Key	Collation	Default value	Policy tags	Description
□	customer_id	STRING	NULLABLE	-	-	-	-	-
□	customer_unique_id	STRING	NULLABLE	-	-	-	-	-
□	customer_zip_code_prefix	INTEGER	NULLABLE	-	-	-	-	-
□	customer_city	STRING	NULLABLE	-	-	-	-	-
□	customer_state	STRING	NULLABLE	-	-	-	-	-

Insights:

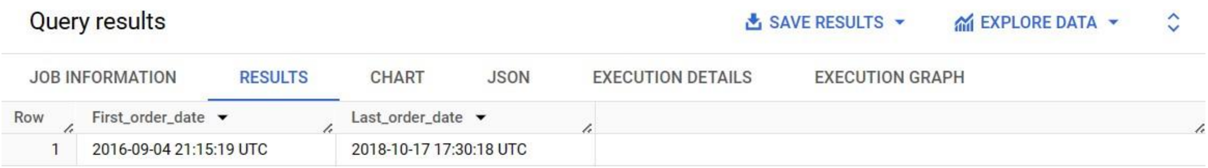
1. All columns in the Customer Table are saved as STRING, except customer_zip_code_prefix which is an INTEGER.
2. It contains Customer details such as Id of the customer, city, state and Zip code respectively.

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

Answer Query:

```
select
     min(order_purchase_timestamp) as First_order_date,
     max(order_purchase_timestamp) as Last_order_date
 from `Target_SQL.orders`
```

Query Results:



Row	First_order_date	Last_order_date
1	2016-09-04 21:15:19 UTC	2018-10-17 17:30:18 UTC

Insights:

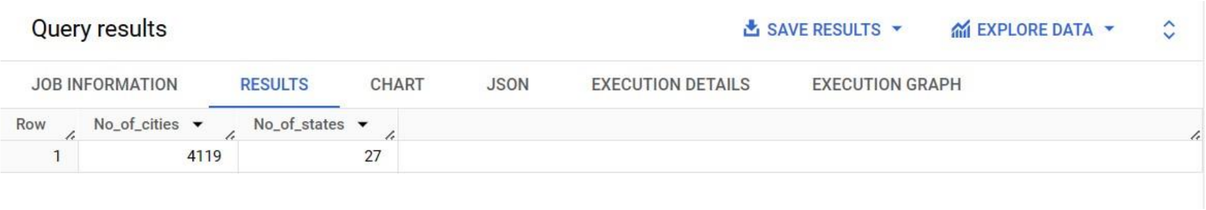
Based on the job results, the orders are placed in Target from the time period 2016 to 2018.

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

Answer Query:

```
select
    count(distinct customer_city) as No_of_cities,
    count(distinct customer_state) as No_of_states
    from `Target_SQL.customers` c inner join `Target_SQL.orders` o
    on c.customer_id = o.customer_id
```

Job Results:



Row	No_of_cities	No_of_states
1	4119	27

Insights:

1. Target dataset which contains 27 States and 4119 Cities.
2. This will give a insights into know about the customer distribution in the states and cities and business operations of Target in Brazil

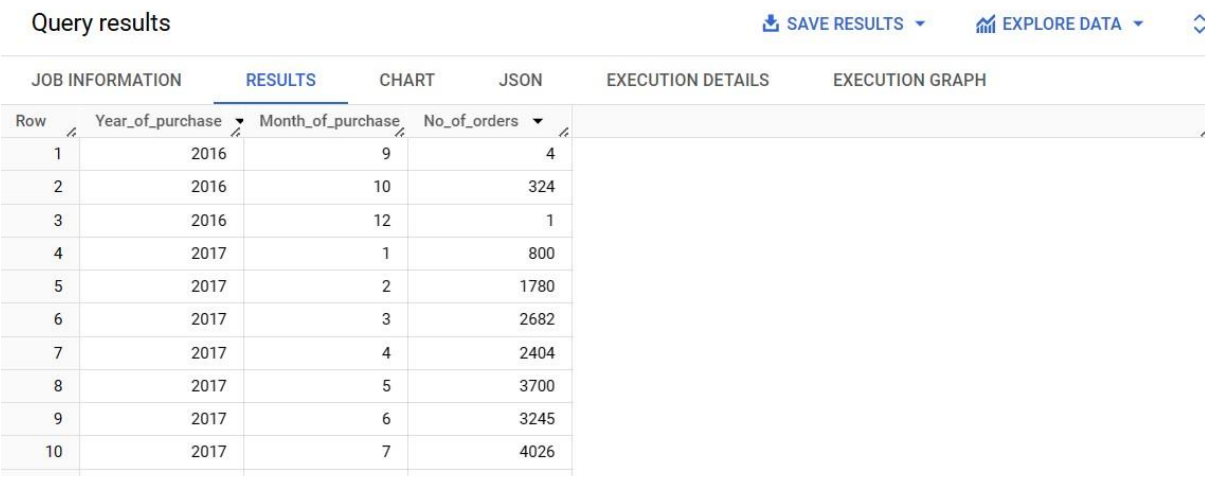
Q2: In-depth Exploration:

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

Answer Query:

```
select extract(year from order_purchase_timestamp) as Year_of_purchase,
       extract(Month from order_purchase_timestamp) as Month_of_purchase,
       count (*) as No_of_orders
       from `Target_SQL.orders`
       group by 1,2
       order by 1,2
```

Job Results:



Row	Year_of_purchase	Month_of_purchase	No_of_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
8	2017	5	3700
9	2017	6	3245
10	2017	7	4026

Graph:

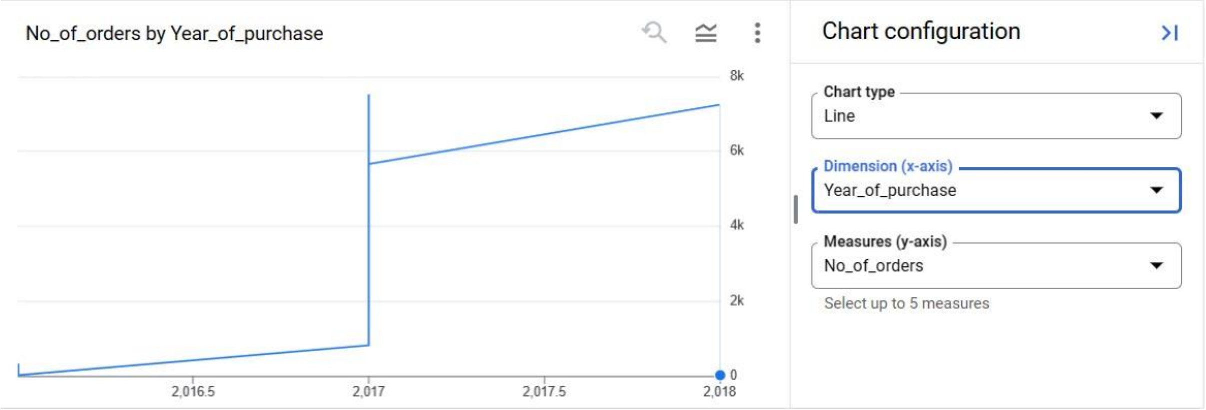


Chart type: Line

Dimension (x-axis): Year_of_purchase

Measures (y-axis): No_of_orders

Select up to 5 measures

Insights:

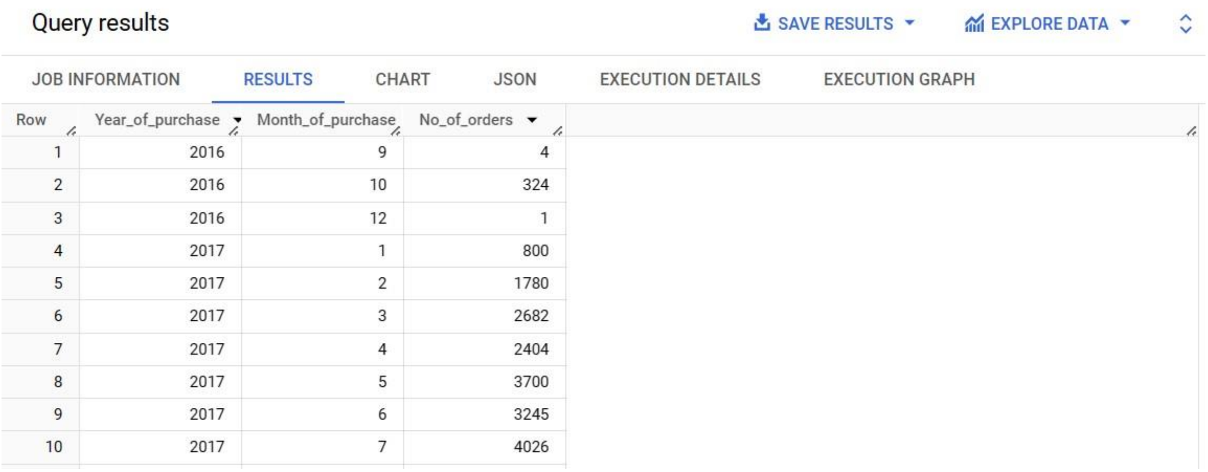
There has been an upward trend in the number of orders over the years from the time period 2016 to 2018 after examining the results.

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

Answer Query:

```
select extract(year from order_purchase_timestamp) as Year_of_purchase,
       extract(month from order_purchase_timestamp) as Month_of_purchase,
       count (*) as No_of_orders
       from `Target_SQL.orders`
       group by 1,2
       order by 1,2
```

Job Results:



Row	Year_of_purchase	Month_of_purchase	No_of_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
8	2017	5	3700
9	2017	6	3245
10	2017	7	4026

Graph:

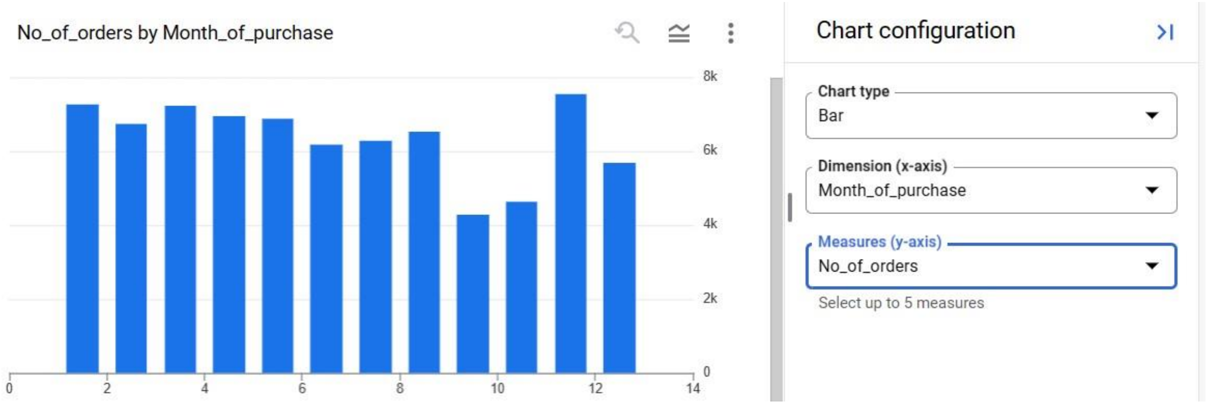


Chart type: Bar

Dimension (x-axis): Month_of_purchase

Measures (y-axis): No_of_orders

Select up to 5 measures

Insights:

1. We see a seasonal trend for Nov 2017 there is a huge increase in the orders placed as people may preorder for Christmas eve.
2. There is also a growing trend in Jan 2017 and Jan 2018 where New Years is approaching and carnival in Feburary.

3. Understanding the seasonality will helps us to understand the customer behaviour which improve the operational planning and marketing tactics. It helps us to plan better in the inventory management, peak period identification and allocation of resources.

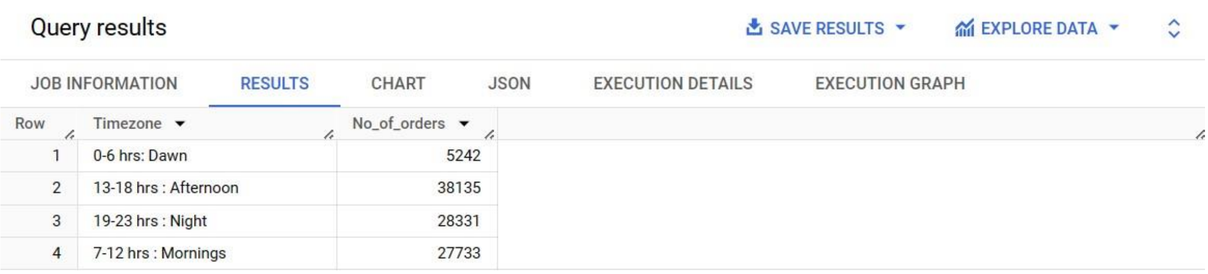
3.During what time of the day, do the Brazilian customers mostly place their orders? (Dawn, Morning, Afternoon or Night)

i. 0-6 hrs : Dawn
ii. 7-12 hrs : Mornings
iii. 13-18 hrs : Afternoon
iv. 19-23 hrs : Night

Answer Query:

```
select
     case
     when extract(hour from order_purchase_timestamp) Between 0 and 6
     then  '0-6 hrs: Dawn'
     when extract(hour from order_purchase_timestamp) Between 7 and 12
     then  '7-12 hrs : Mornings'
     when extract(hour from order_purchase_timestamp) Between 13 and 18
     then  '13-18 hrs : Afternoon'
     when extract(hour from order_purchase_timestamp) Between 19 and 23
     then  '19-23 hrs : Night'
     else
     'Invalid'
     end as Timezone ,
     count(*) as No_of_orders
 from `Target_SQL.orders`
 group by 1
 order by 1
```

JobResults:



Row	Timezone	No_of_orders
1	0-6 hrs: Dawn	5242
2	13-18 hrs : Afternoon	38135
3	19-23 hrs : Night	28331
4	7-12 hrs : Mornings	27733

Insights:

1. Based on the order purchased timestamp, we can find out the hourly placed orders and divided them into timely groups Dawn, Morning, Afternoon and Night.
2. The results are sorted based on the orders placed in the given time period.
3. From the results, **Brazilian customers placed more orders in the afternoon.**

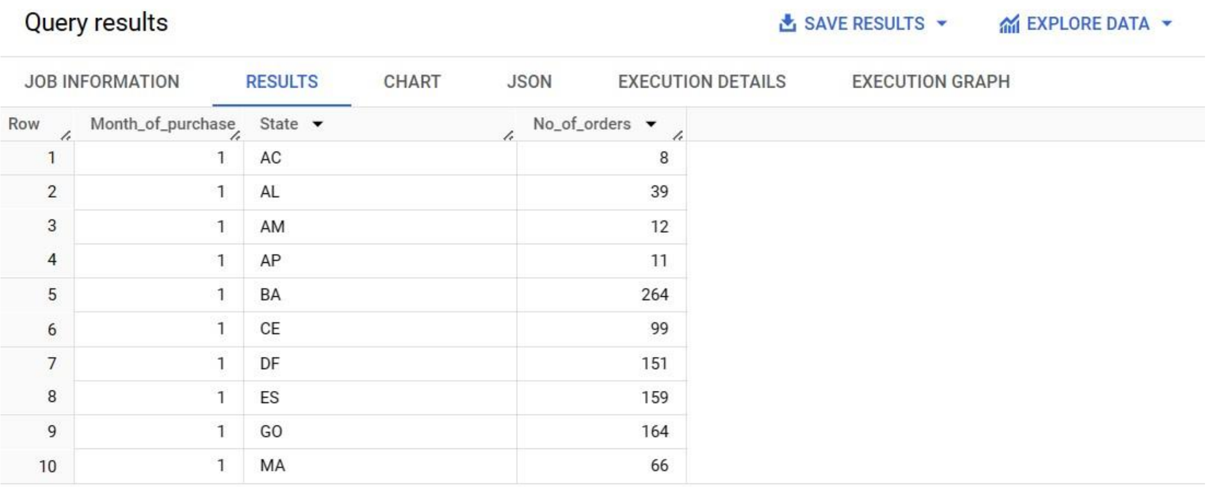
Q3: Evolution of E-commerce orders in the Brazil region:

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

Answer Query:

```
select
    extract(month from O.order_purchase_timestamp) as Month_of_purchase,
    C.customer_state as State,
    count (*) as No_of_orders
    FROM `Target_SQL.orders` O
    inner join `Target_SQL.customers` C on O.customer_id = C.customer_id
    group by 1,2
    order by 1,2
```

Job Results:



Row	Month_of_purchase	State	No_of_orders
1	1	AC	8
2	1	AL	39
3	1	AM	12
4	1	AP	11
5	1	BA	264
6	1	CE	99
7	1	DF	151
8	1	ES	159
9	1	GO	164
10	1	MA	66

Graph:

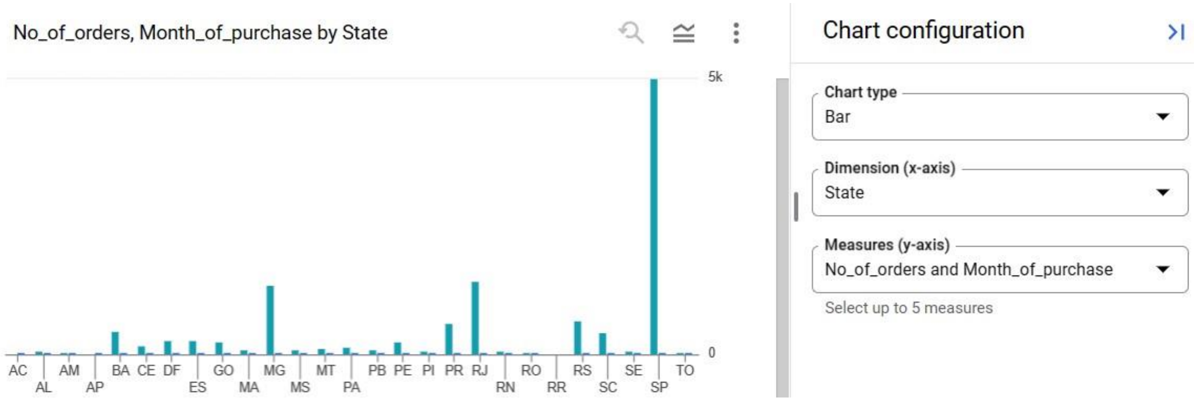


Chart configuration

Chart type: Bar

Dimension (x-axis): State

Measures (y-axis): No_of_orders and Month_of_purchase

Select up to 5 measures

Insights:

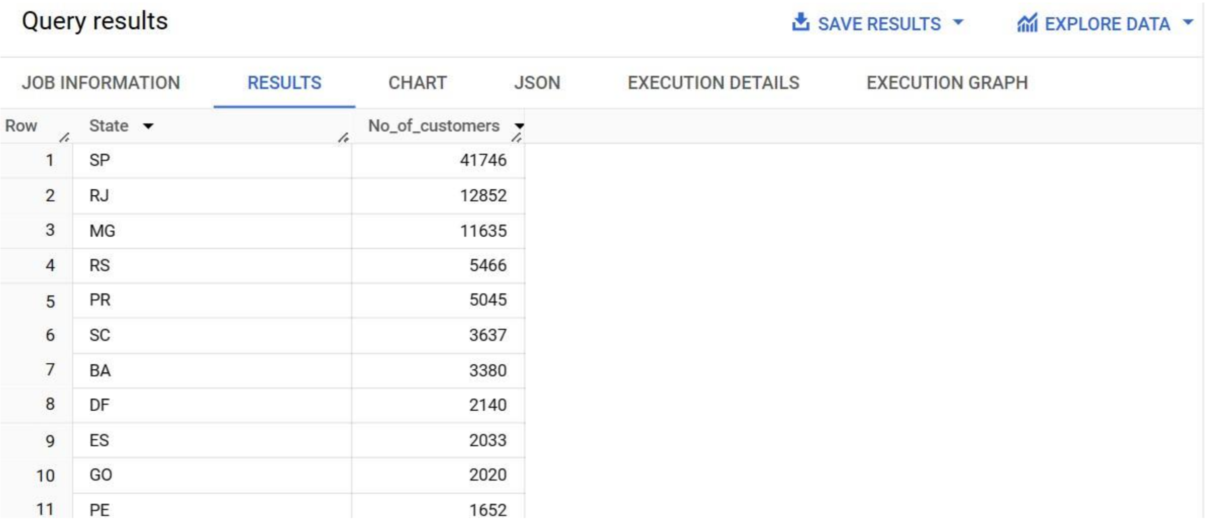
1. We can learn more about the monthly order count for each state by examining the query's results. In our data, we can find that for every month the state called **SP has the highest number of orders.**
2. We can target marketing efforts in states with rising order volumes and inventory management based on order trends across different states by analysing these insights.

2.How are the customers distributed across all the states?

Answer Query:

```
select customer_state as State,
       count(Distinct customer_id) as No_of_customers
       from `Target_SQL.customers`
       group by 1
       order by 2 desc
```

Job Results:



Row	State	No_of_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

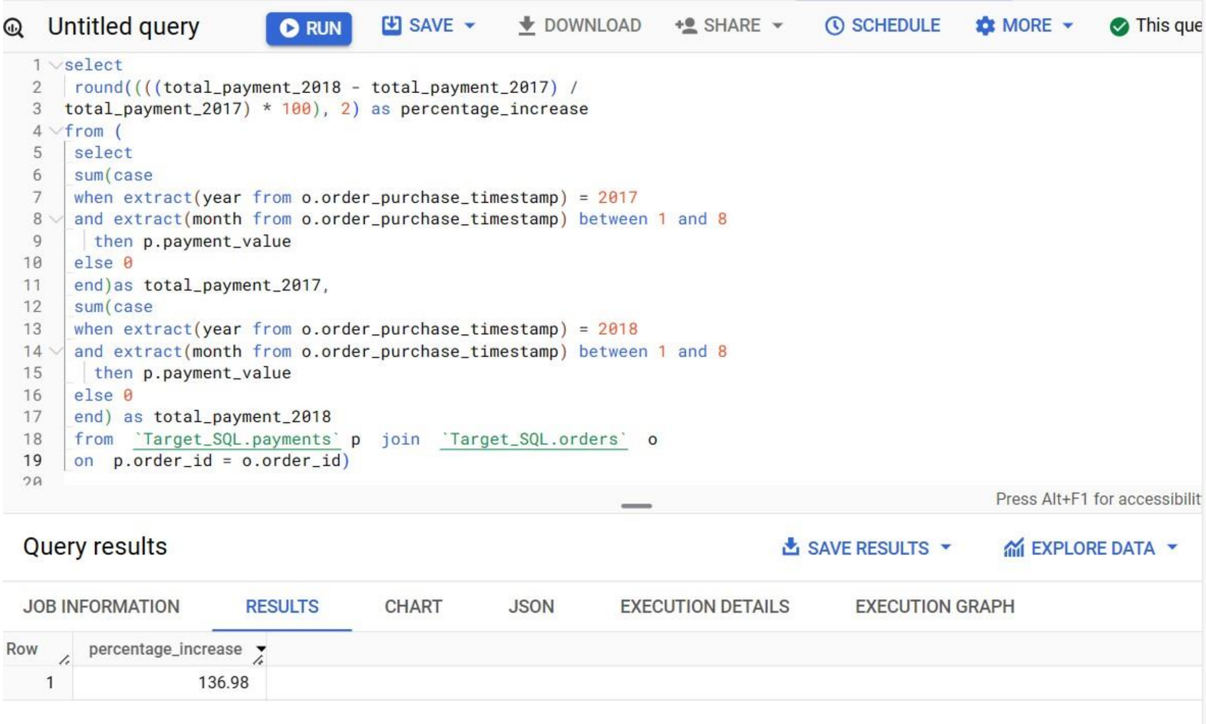
Insights:

1. The distribution of customers across states will be shown by analysing the query's results.
2. The states with largest and fewest customer database can be determined.
3. From the results**, SP has the highest Customers and RR has the least Customers**.
4. It helps us to make good business decisions by looking at the customer distribution between the states.

Q4: Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.

1. Get the % increase in the cost of orders from year 2017 to 2018 (include months between Jan to Aug only). You can use the "payment_value" column in the payments table to get the cost of orders.

Answer Query and Results:



```
select
 round((((total_payment_2018 - total_payment_2017) /
total_payment_2017) * 100), 2) as percentage_increase
from (
 select
 sum(case
 when extract(year from o.order_purchase_timestamp) = 2017
 and extract(month from o.order_purchase_timestamp) between 1 and 8
   then p.payment_value
 else 0
 end)as total_payment_2017,
 sum(case
 when extract(year from o.order_purchase_timestamp) = 2018
 and extract(month from o.order_purchase_timestamp) between 1 and 8
   then p.payment_value
 else 0
 end) as total_payment_2018
 from  `Target_SQL.payments` p  join  `Target_SQL.orders`  o
 on  p.order_id = o.order_id)
```

Row	percentage_increase
1	136.98

Insights:

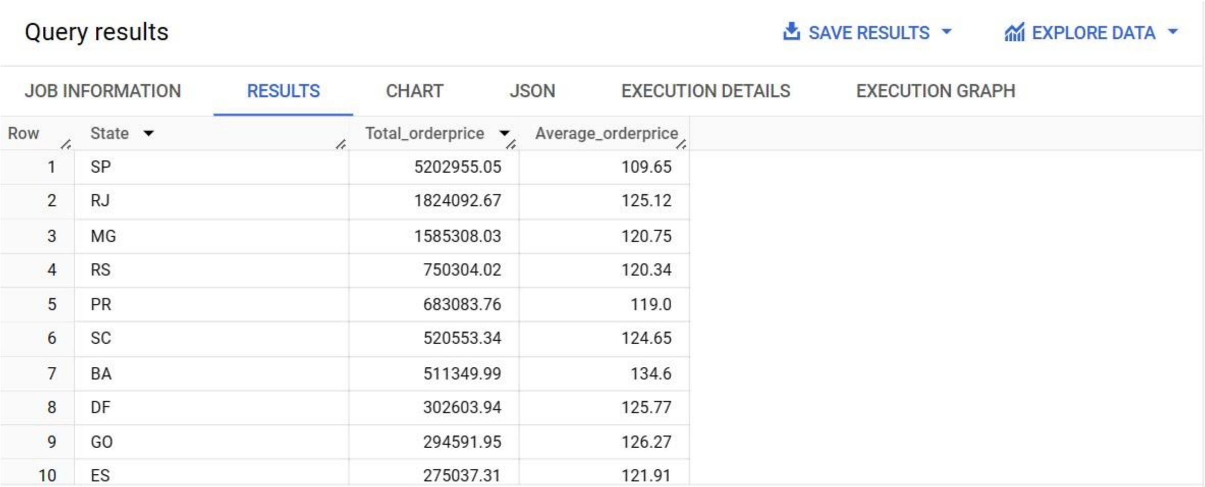
1. From 2017 to 2018, only the orders placed during the month Jan to Aug are considered.
2. Based on the payment value, % of increase in the months 1-8 over the year is calculated.
3. From the Query results, we find out **the growth rate of 137%** increase over the year 2017-2018.

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

Answer Query:

```
Select C.customer_state as State,
       round(Sum(OI.price),2) as Total_orderprice,
       round(Avg(OI.price),2) as Average_orderprice
       from `Target_SQL.customers` C
       inner join `Target_SQL.orders`O on C.customer_id = O.customer_id
       inner join `Target_SQL.order_items`OI on O.order_id = OI.order_id
 group by C.customer_state
 order by Total_orderprice desc
```

Job Results:



Row	State	Total_orderprice	Average_orderprice
1	SP	5202955.05	109.65
2	RJ	1824092.67	125.12
3	MG	1585308.03	120.75
4	RS	750304.02	120.34
5	PR	683083.76	119.0
6	SC	520553.34	124.65
7	BA	511349.99	134.6
8	DF	302603.94	125.77
9	GO	294591.95	126.27
10	ES	275037.31	121.91

Insights:

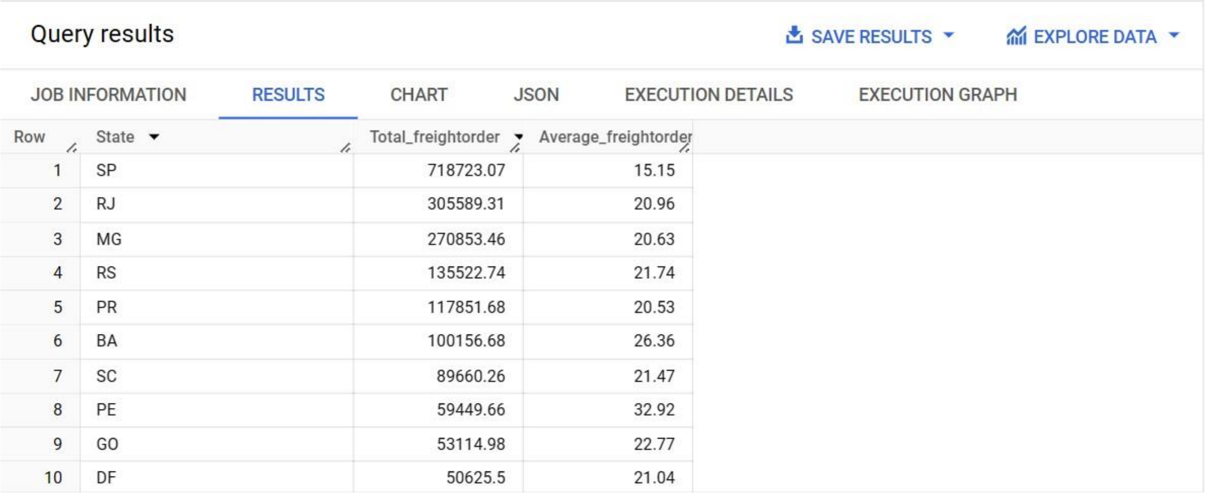
1. The sum of all order prices for each state is displayed in the "Total_orderprice" column, which represents the total amount of orders placed.
2. The "Average_orderprice" column shows the average order price for that state.
3. By analysing the results, we can find out the states with highest and least order values.

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

Answer Query:

```
Select C.customer_state as State,
       round(Sum(OI.freight_value),2) as Total_freightorder,
       round(Avg(OI.freight_value),2) as Average_freightorder
       from `Target_SQL.customers` C
       inner join `Target_SQL.orders`O on C.customer_id = O.customer_id
       inner join `Target_SQL.order_items`OI on O.order_id = OI.order_id
       group by C.Customer_state
       order by Total_freightorder desc
```

Job Results:



Row	State	Total_freightorder	Average_freightorder
1	SP	718723.07	15.15
2	RJ	305589.31	20.96
3	MG	270853.46	20.63
4	RS	135522.74	21.74
5	PR	117851.68	20.53
6	BA	100156.68	26.36
7	SC	89660.26	21.47
8	PE	59449.66	32.92
9	GO	53114.98	22.77
10	DF	50625.5	21.04

Insights:

1. The sum of all freight order prices for each state is displayed in the "Total_freightorder" column, which represents the total amount of freight orders.
2. The "Average_freightorder" column shows the average freight order for that state.
3. By analysing the results, we can find out the states with highest and least freight order values.

Q5: Analysis based on sales, freight and delivery time.

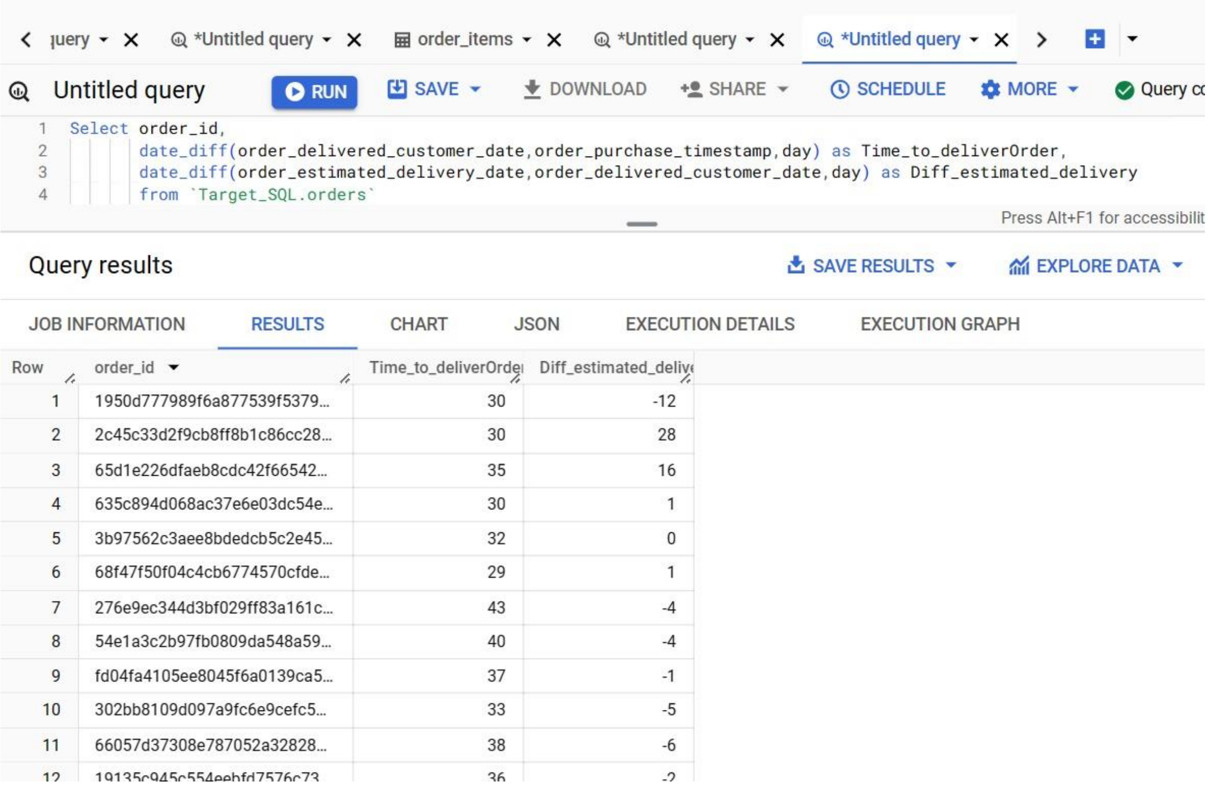
1. Find the no. of days taken to deliver each order from the order's purchase date as delivery time.

Also, calculate the difference (in days) between the estimated & actual delivery date of an order. Do this in a single query.

You can calculate the delivery time and the difference between the estimated & actual delivery date using the given formula:

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

Answer and Job Results:



```
Select order_id,
        date_diff(order_delivered_customer_date,order_purchase_timestamp,day) as Time_to_deliverOrder,
        date_diff(order_estimated_delivery_date,order_delivered_customer_date,day) as Diff_estimated_delivery
        from `Target_SQL.orders`
```

Row	order_id	Time_to_deliverOrde	Diff_estimated_delive
1	1950d777989f6a877539f5379...	30	-12
2	2c45c33d2f9cb8ff8b1c86cc28...	30	28
3	65d1e226dfaeb8cdc42f66542...	35	16
4	635c894d068ac37e6e03dc54e...	30	1
5	3b97562c3aee8bdedcb5c2e45...	32	0
6	68f47f50f04c4cb6774570cfde...	29	1
7	276e9ec344d3bf029ff83a161c...	43	-4
8	54e1a3c2b97fb0809da548a59...	40	-4
9	fd04fa4105ee8045f6a0139ca5...	37	-1
10	302bb8109d097a9fc6e9cefc5...	33	-5
11	66057d37308e787052a32828...	38	-6
12	19135c945c554eebfd7576c73	36	-2

Insights:

1. By analysing the delivery_time and diff_estimated_delivery columns, early delivery and any delays in the delivery process can be found.
2. These columns can be used to find out the elements that affect delivery times or discrepancies between estimated and actual delivery dates.

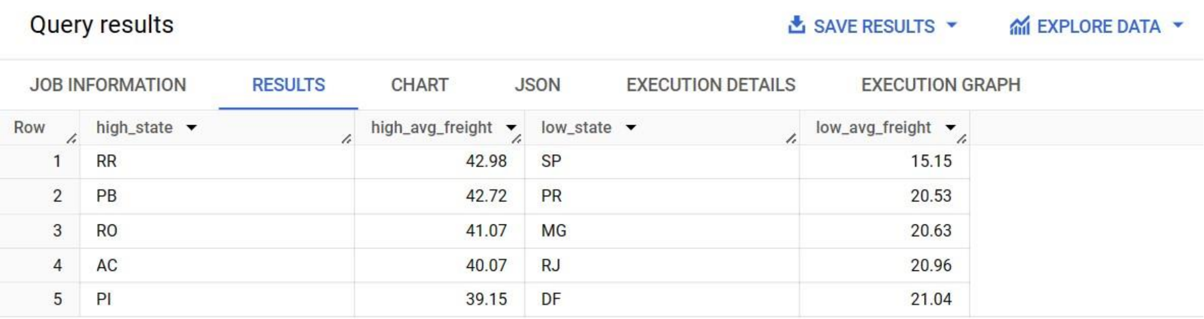
2.Find out the top 5 states with the highest & lowest average freight value.

Answer Query:

```
select
 high.customer_state as high_state,
 high.average_freight_value as high_avg_freight,
 low.customer_state as low_state,
 low.average_freight_value as low_avg_freight
from
(
 select
 c.customer_state,
 round(avg(p.freight_value),2) as average_freight_value,
 row_number() over(order by
(round(avg(p.freight_value),2))desc) AS rowval1
 from `Target_SQL.orders` o
 join `Target_SQL.order_items` p on o.order_id = p.order_id
 join `Target_SQL.customers` c on o.customer_id = c.customer_id
 group by c.customer_state
 order by average_freight_value desc
limit  5
) as high

join
(
 select
 c.customer_state,
 round(avg(p.freight_value),2) as average_freight_value,
 row_number() over(order by (round(avg(p.freight_value),2))) as rowval2
 from `Target_SQL.orders` o join `Target_SQL.order_items` p
 on o.order_id = p.order_id
 join `Target_SQL.customers` as c on o.customer_id = c.customer_id
 group by c.customer_state
 order by average_freight_value
limit 5
 ) as low
 on high.rowval1 = low.rowval2
```

Job Results:



Row	high_state	high_avg_freight	low_state	low_avg_freight
1	RR	42.98	SP	15.15
2	PB	42.72	PR	20.53
3	RO	41.07	MG	20.63
4	AC	40.07	RJ	20.96
5	PI	39.15	DF	21.04

Insights:

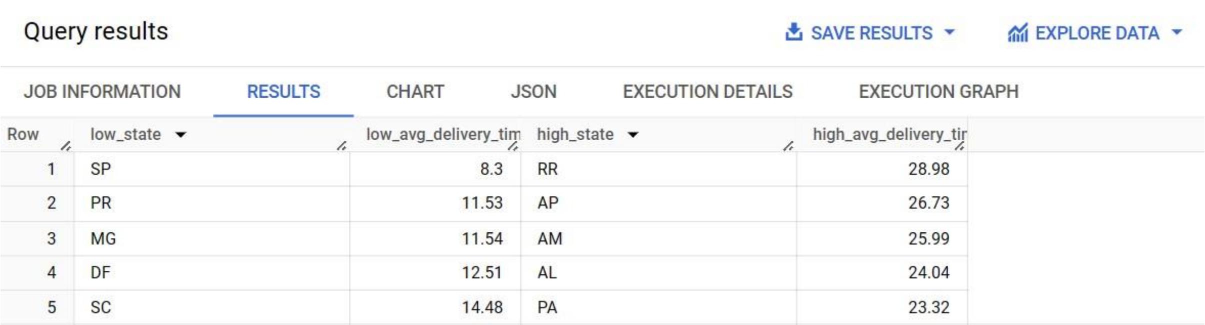
1. The states with the highest average freight values like states called RR and PB may experience greater shipping prices due to reasons like remote locations, higher transportation costs, or supply chain difficulties.
2. It might be useful to optimize logistics operations to locate places with relatively reduced shipping prices by looking at the states with the lowest average freight values like states such as SP and PR.

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

Answer Query:

```
WITH cte AS
(
 select
 c.customer_state,
 round(avg(t1.delivery_time),2) as avg_delivery_time
 from
 (
 select
 *,
 timestamp_diff(order_delivered_customer_date,order_purchase_timestamp, day) as
delivery_time,
 from `Target_SQL.orders` where order_status = 'delivered'and
order_delivered_customer_date IS NOT NULL
 order by order_purchase_timestamp
) as t1
join
`Target_SQL.customers` as c on t1.customer_id = c.customer_id
group by c.customer_state
order by avg_delivery_time
)
select
 c1.customer_state as low_state,
 c1.avg_delivery_time as low_avg_delivery_time,
 c2.customer_state as high_state,
 c2.avg_delivery_time as high_avg_delivery_time
from
(
select
 *,
 row_number() over (order by cte.avg_delivery_time desc) as rowval2
 from cte
 order by rowval2
) as c2
join
(
 select
 *,
 row_number() over(order by cte.avg_delivery_time) as rowval1
 from cte
 order by rowval1
) as c1
on c1.rowval1 = c2.rowval2
limit 5
```

Job Results:



Row	low_state	low_avg_delivery_tim	high_state	high_avg_delivery_tir
1	SP	8.3	RR	28.98
2	PR	11.53	AP	26.73
3	MG	11.54	AM	25.99
4	DF	12.51	AL	24.04
5	SC	14.48	PA	23.32

Insights:

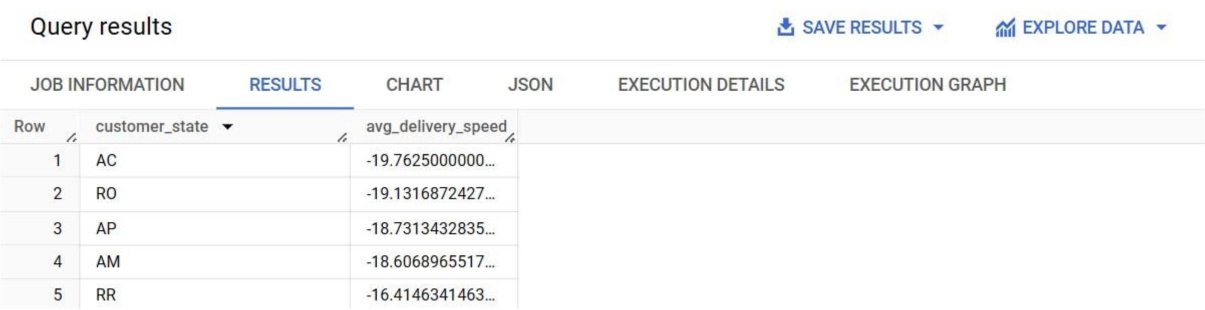
Finding areas with effective delivery operations can be done by looking at the states like SP and PR with the lowest average delivery times and states called RR and AP with highest average delivery times.

4.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.

Answer Query:

```
with delivery_speed as (
select
 c.customer_state,
 avg(date_diff(o.order_delivered_customer_date,o.order_estimated_delivery_da
te,day)) as avg_delivery_speed,
 row_number() over(order by
avg(date_diff(o.order_delivered_customer_date,o.order_estimated_delivery_dat
e,day))) as fastest_rank
 from `Target_SQL.orders` as o join `Target_SQL.customers` as C
 on o.customer_id = c.customer_id
 where o.order_delivered_customer_date IS NOT NULL and
o.order_estimated_delivery_date IS NOT NULL
 group by c.customer_state
)
select customer_state, avg_delivery_speed
from delivery_speed
where fastest_rank <= 5
order by avg_delivery_speed
```

Job Results:



Row	customer_state	avg_delivery_speed
1	AC	-19.7625000000...
2	RO	-19.1316872427...
3	AP	-18.7313432835...
4	AM	-18.6068965517...
5	RR	-16.4146341463...

Insights:

1. Based on the results, we can find out the top 5 states with fastest delivery speed.
2. This data helps us to improve the market standards with the track record of order delivery

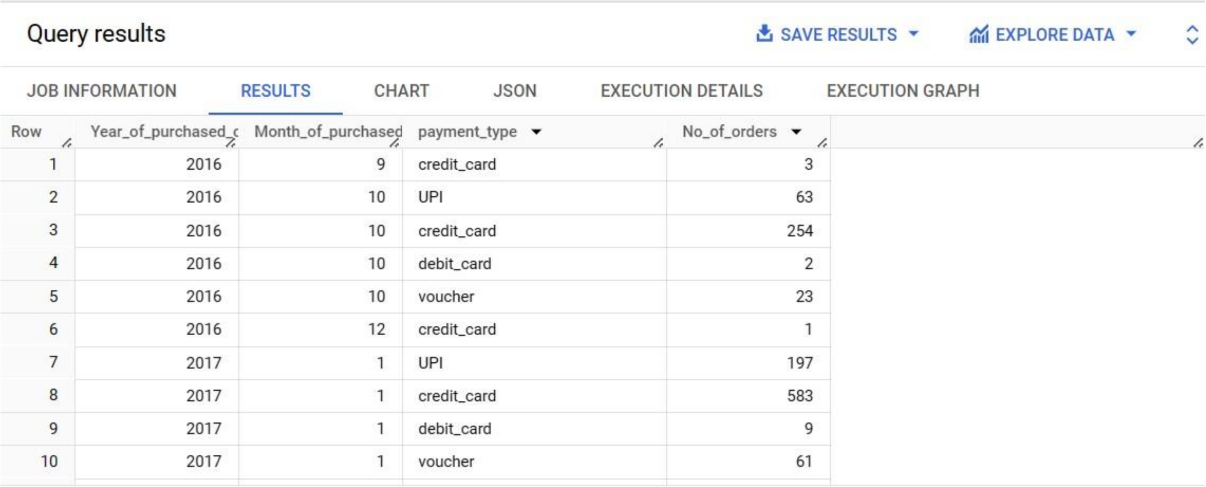
Q6: Analysis based on the payments:

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

Answer Query:

```
select extract(Year from o.order_purchase_timestamp) as Year_of_purchased_order,
       extract(Month from o.order_purchase_timestamp)as Month_of_purchased_order,
       P.payment_type,
       count (*) as No_of_orders
       from `Target_SQL.payments`P
       inner join `Target_SQL.orders`O
       on P.order_id = O.order_id
       group by 1,2,3
       order by 1,2,3
```

Job Results:



Row	Year_of_purchased_c	Month_of_purchased	payment_type	No_of_orders
1	2016	9	credit_card	3
2	2016	10	UPI	63
3	2016	10	credit_card	254
4	2016	10	debit_card	2
5	2016	10	voucher	23
6	2016	12	credit_card	1
7	2017	1	UPI	197
8	2017	1	credit_card	583
9	2017	1	debit_card	9
10	2017	1	voucher	61

Insights:

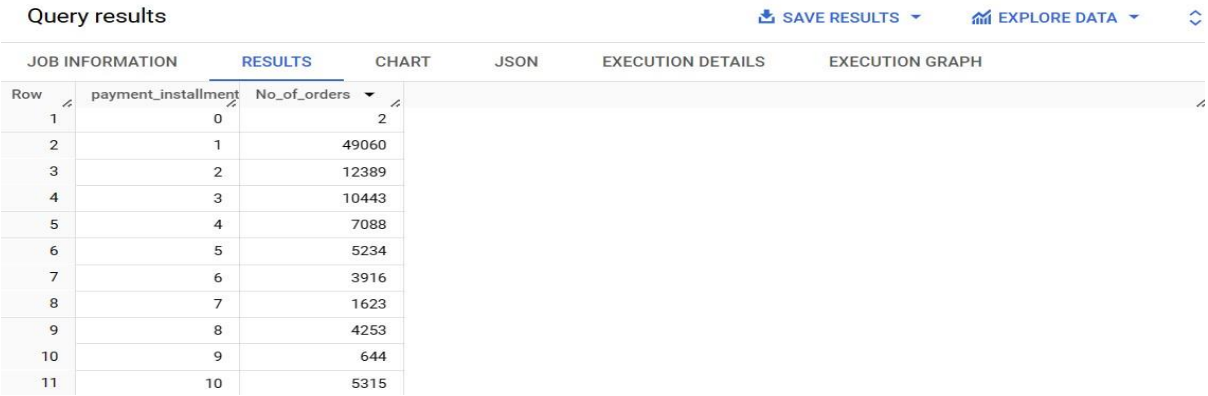
1. Based on the payment type preferences noticed on each month, helps to know the customers most preferred payment types.
2. From the data results, we found out the Credit card payment type is most used in Nov2017.

2.Find the no. of orders placed on the basis of the payment instalments that have been paid.

Answer Query:

```
Select payment_installments,
       count(distinct order_id) as No_of_orders
       from `Target_SQL.payments`
       group by 1
       order by 1
```

Job Results:



Row	payment_installment	No_of_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
10	9	644
11	10	5315

Insights:

From the given data, we found out 49060 orders are placed where the Payment instalment is 1.

Business Recommendations:

- *Enhance customer experience*: Target can offer personalized recommendations based on customer purchase history and browsing behaviour. The company can also provide efficient customer support through live chat, email, and phone calls. Additionally, Target can improve the online shopping experience by making the website and mobile app more user-friendly.

- *Optimize pricing strategies:* Target can analyse pricing trends and competition in the market to ensure competitive pricing while maintaining profitability. The company can also implement dynamic pricing strategies that consider factors like customer demand, product popularity, and seasonality.

- *Streamline logistics and delivery*: Target can improve logistics and delivery processes by optimizing warehouse operations, partnering with reliable shipping carriers, and leveraging technology solutions like real-time tracking.

- *Localize marketing and promotions*: The company can understand the specific needs and preferences of customers in different regions of Brazil and develop targeted marketing strategies to effectively reach and engage with them.