TITLE:Business Case Study - Target SQL

Description: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.

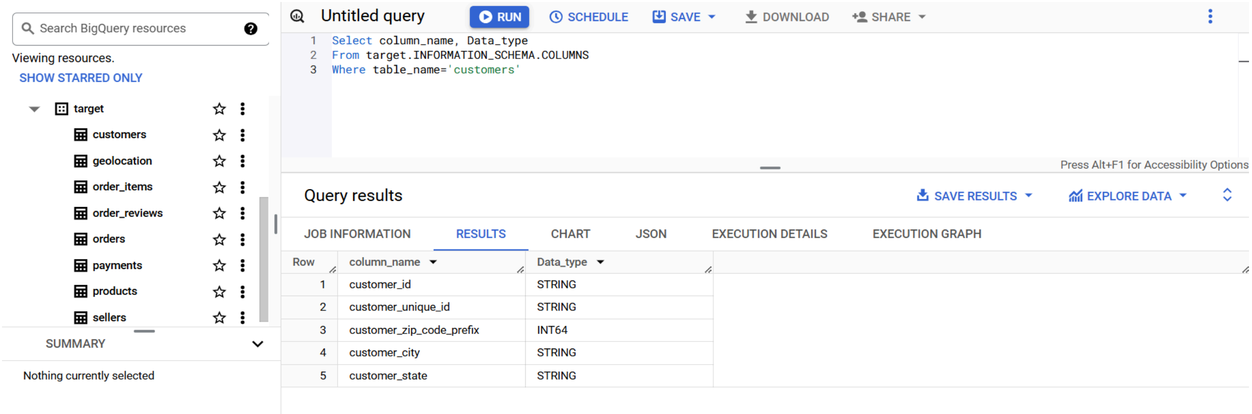
By analyzing this extensive dataset, it becomes possible to gain valuable insights into Target's operations in Brazil. The information can shed light on various aspects of the business, such as order processing, pricing strategies, payment and shipping efficiency, customer demographics, product characteristics, 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.

```
Select column_name, Data_type
From target.INFORMATION_SCHEMA.COLUMNS
Where table_name='customers'
```

Output:



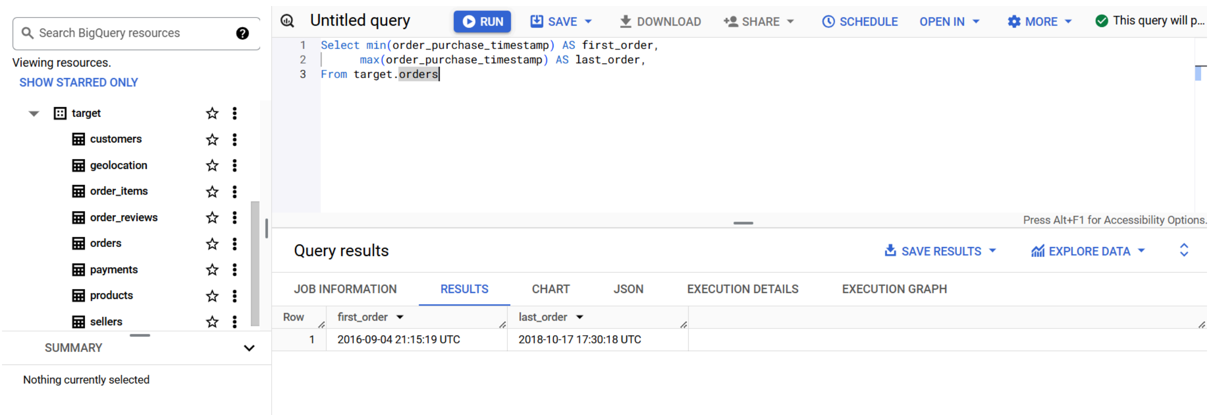
Insights:

In the above screenshot we can see all the datatypes are of string format except customer_zip_code_prefix which is Integer.

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

```
Select min(order_purchase_timestamp) AS first_order,
        max(order_purchase_timestamp) AS last_order,
From target.orders
```

Output:



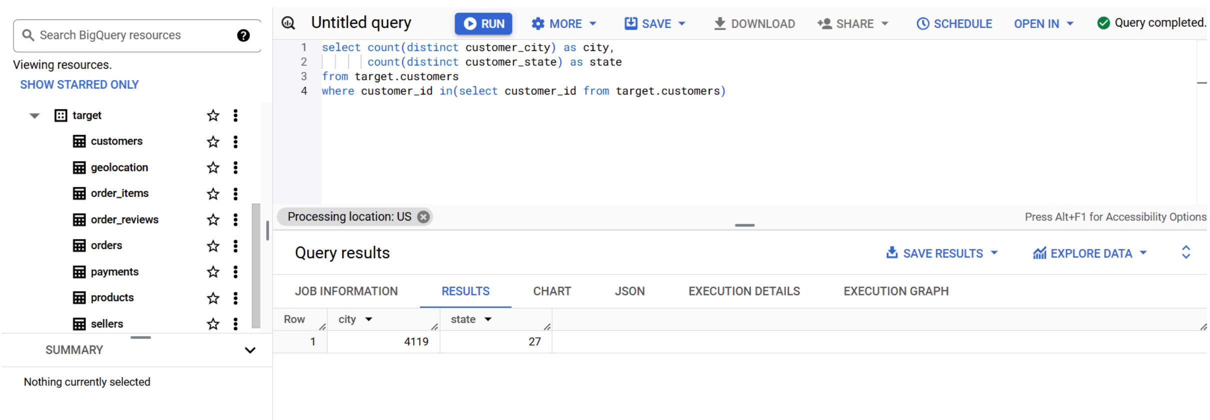
Insights:

The time range between which the orders were placed is 2 years.First column is timestamp of first order placed and second column is timestamp of last order placed in the given range.

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

```
select count(distinct customer_city) as city,
        count(distinct customer_state) as state
from target.customers
where customer_id in(select customer_id from target.customers)
```

Output:



Insights:

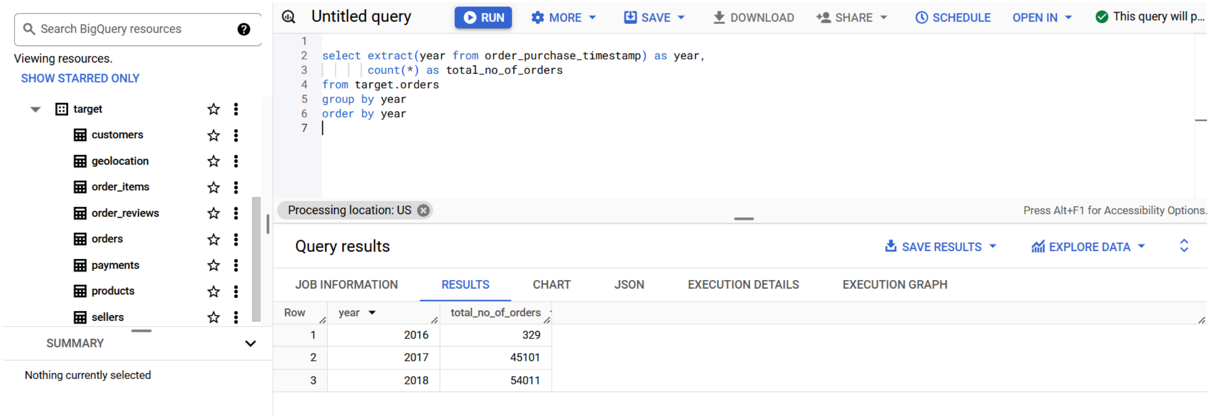
Total 4119 number of cities and 27 states of customers ordered in the given period of time. In the above query we have used the sub query for finding the customer_id's of customers who ordered during the given period.

Q2. In-depth Exploration:

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

```
select extract(year from order_purchase_timestamp) as year,
          count(*) as total_no_of_orders
from target.orders
group by year
order by year
```

Output:



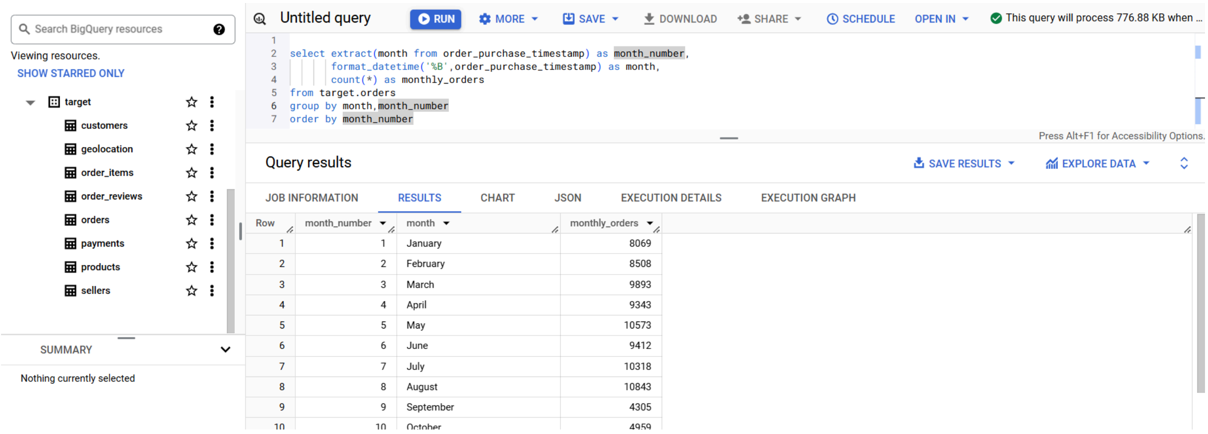
Insights:

In the above output we can see the number of orders placed every year. After analysing the output there is a growing trend in the number of orders placed over the past year.

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

```
select extract(month from order_purchase_timestamp) as month_number,
        format_datetime('%B',order_purchase_timestamp) as month,
        count(*) as monthly_orders
from target.orders
group by month,month_number
order by month_number
```

Output:



Row	month_number	month	monthly_orders
1	1	January	8069
2	2	February	8508
3	3	March	9893
4	4	April	9343
5	5	May	10573
6	6	June	9412
7	7	July	10318
8	8	August	10843
9	9	September	4305
10	10	October	4959

Insights:

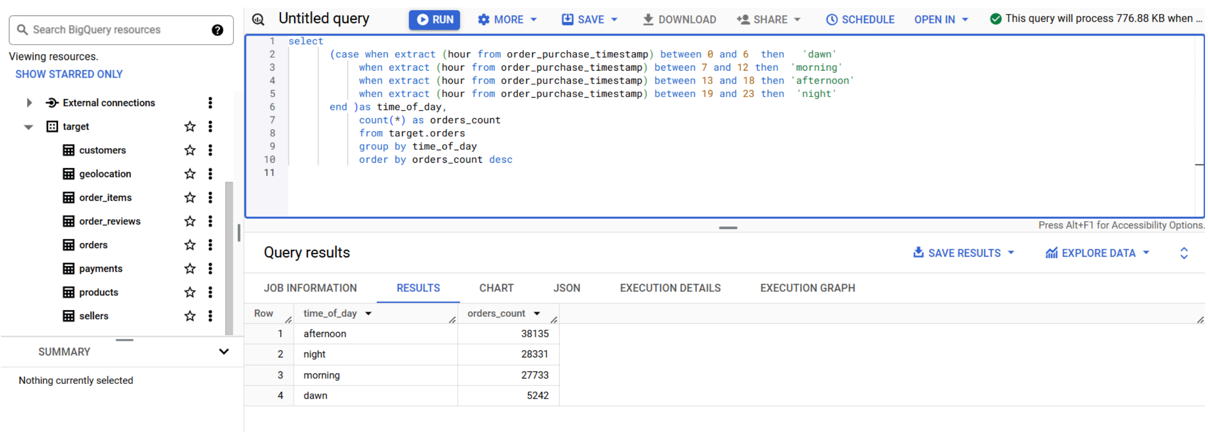
In the above output screenshot we can see some kind of monthly seasonality in terms of the number of orders being placed.
In the month of August highest number of orders were placed. And in the month of September lowest number of orders were placed.

3. 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

```
select
      (case when extract (hour from order_purchase_timestamp) between 0 and 6  then
'dawn'
           when extract (hour from order_purchase_timestamp) between 7 and 12 then
'morning'
           when extract (hour from order_purchase_timestamp) between 13 and 18 then
'afternoon'
           when extract (hour from order_purchase_timestamp) between 19 and 23 then
'night'
     end ) as time_of_day,
         count(*) as orders_count
 from target.orders
 group by time_of_day
 order by orders_count desc
```

Output:



Insights:

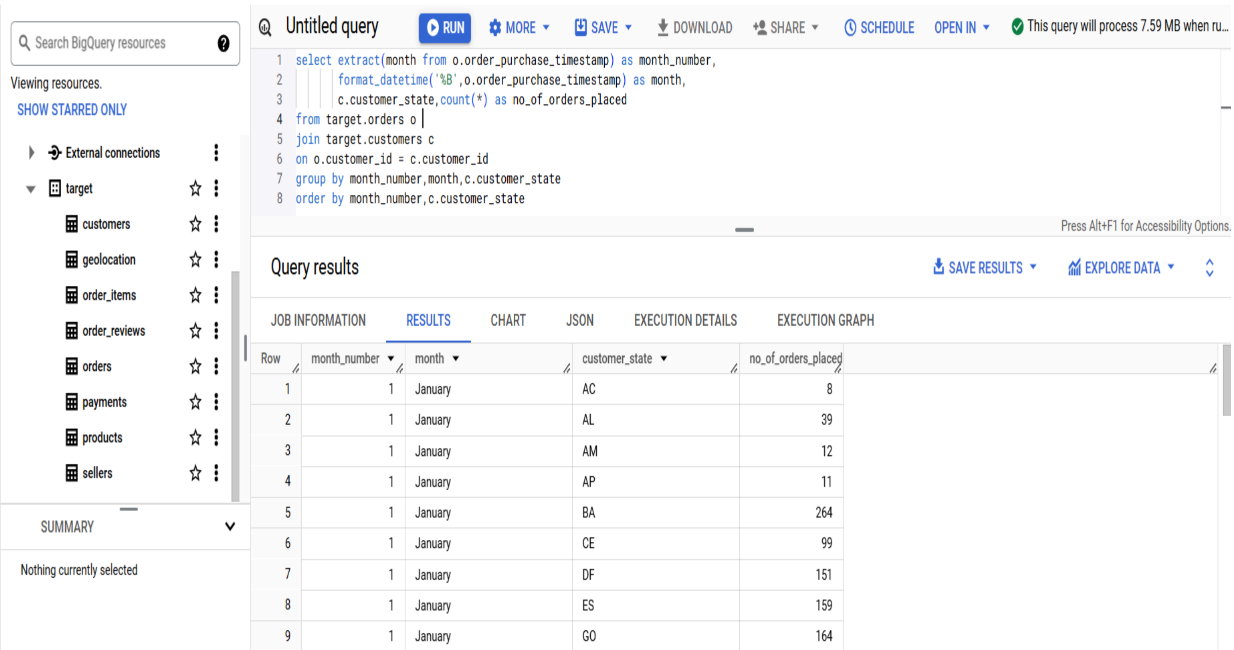
Based on the hour component of the timestamp, the query divides the order timestamps into various time groups (Dawn, Morning, Afternoon, Night). The results are then sorted based on how many orders fell inside each time frame.Brazilian customers frequently order more in the afternoons and customers placed least orders in Dawn time.

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

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

```
select extract(month from o.order_purchase_timestamp) as month_number,
       format_datetime('%B',o.order_purchase_timestamp) as month,
       c.customer_state,count(*) as no_of_orders_placed
from target.orders o
join target.customers c
on o.customer_id = c.customer_id
group by month_number,month,c.customer_state
order by month_number,c.customer_state
```

Output:



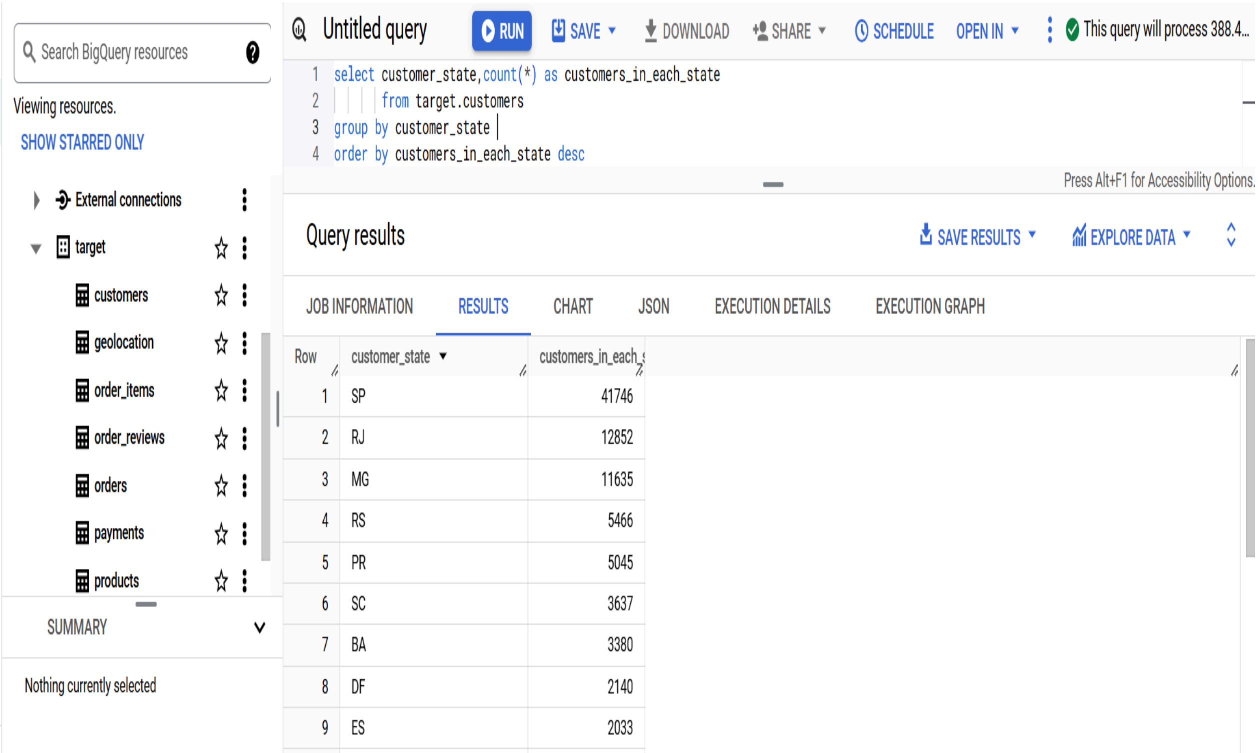
Insights:

In the above output we can find the month on month number of orders placed in each state. After analysing the query result we can find that for every month the state called “SP” has the highest number of orders.

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

```
select customer_state,count(*) as customers_in_each_state
        from target.customers
group by customer_state
order by customers_in_each_state desc
```

Output:



Insights:

In the above query result we can see the total customers of each state along with their states. After analysing the query result we can find that the state called “SP” has the highest number of customers and the state called “RR” has the fewest number of customers.

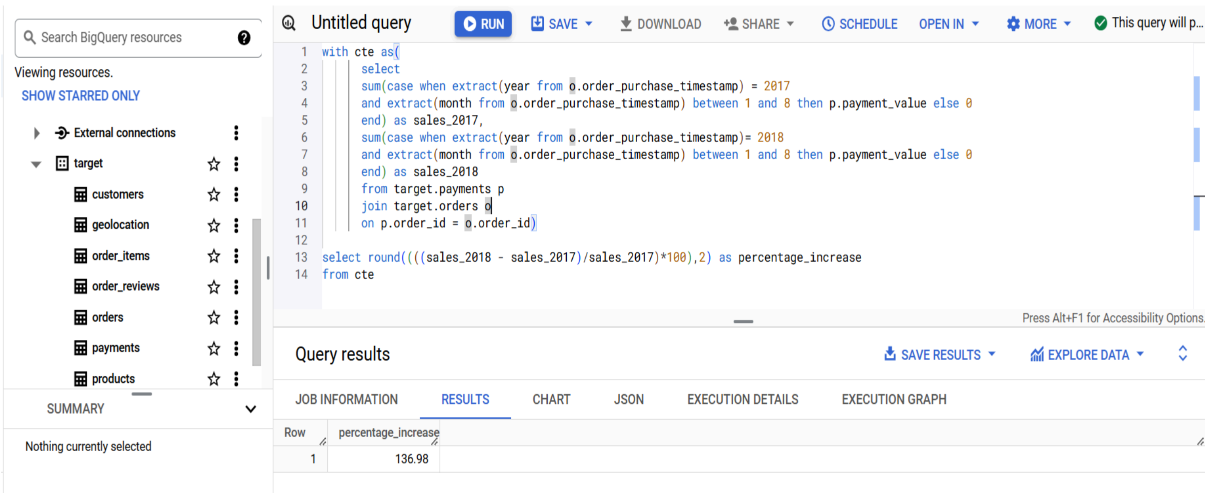
Q4. Impact on Economy: Analyze the money movement by e-commerce by looking at order prices, freight and others.

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.

```
With cte as(
    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 end) as sales_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 sales_2018
    from target.payments p
    join target.orders o
   on p.order_id = o.order_id)

 select round((((sales_2018 - sales_2017)/sales_2017)*100),2) as percentage_increase
 from cte
```

<u>Output:</u>



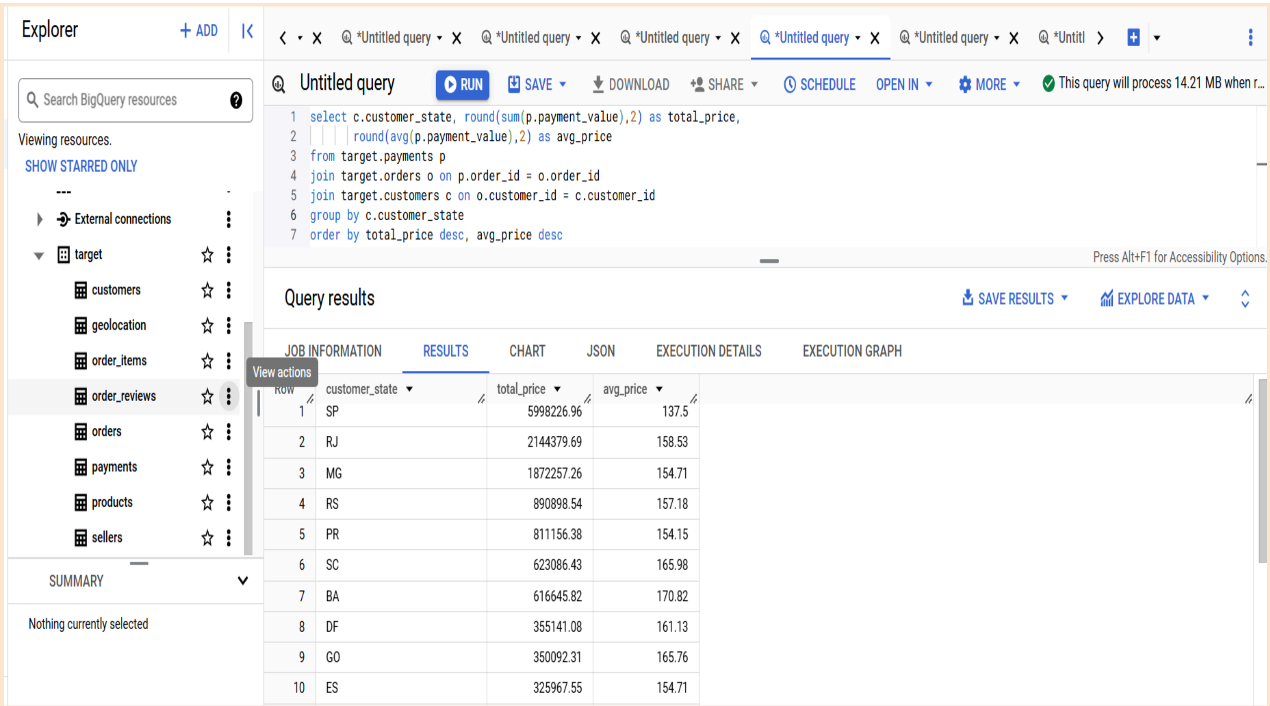
Insights:

In the above query first we have written cte for finding the cost of orders placed from January to August in the year 2017 and 2018 individually.After analysing the result we can find the growth rate approximately 137% from 2017 to 2018.

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

```
select c.customer_state, round(sum(p.payment_value),2) as total_price,
       round(avg(p.payment_value),2) as avg_price
from target.payments p
join target.orders o on p.order_id = o.order_id
join target.customers c on o.customer_id = c.customer_id
group by c.customer_state
order by total_price desc, avg_price desc
```

Output:



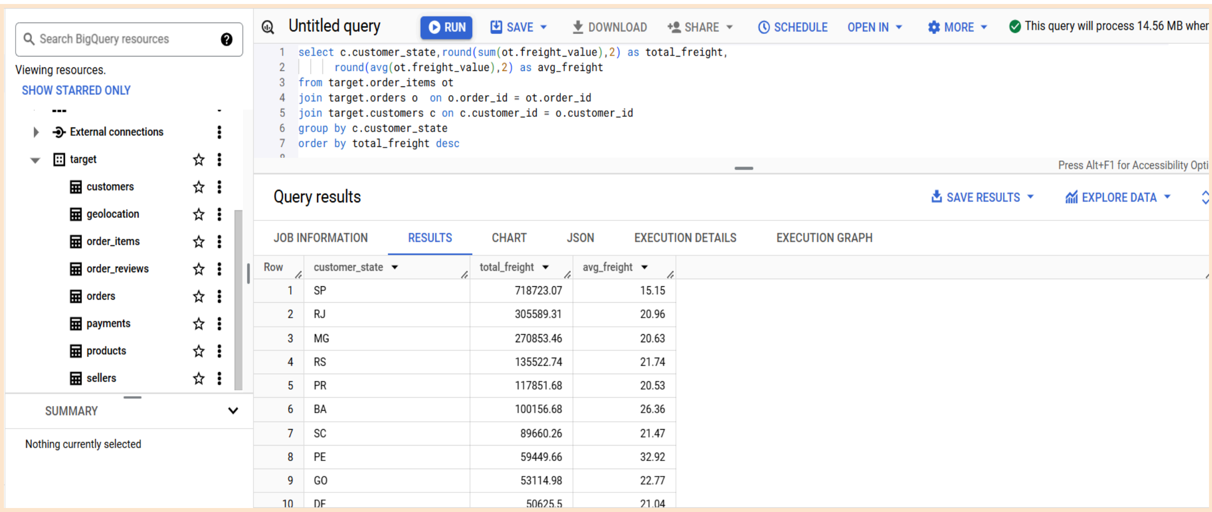
Insights:

In the above result the sum of all order prices for each state is displayed in the "total_order_price" column, which represents the total revenue in each state.The "average_order_price" column shows the average order price of customers in each state.
After analysing the query result we can find that the state called “SP” has the highest revenue and the state called “RR” has the lowest revenue.In the state of “RR” the number of customers are less compared to other states.while increasing the customers we can get more revenue because “RR” state has good average_order_price.

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

```
select c.customer_state,round(sum(ot.freight_value),2) as total_freight,
       round(avg(ot.freight_value),2) as avg_freight
from target.order_items ot
join target.orders o  on o.order_id = ot.order_id
join target.customers c on c.customer_id = o.customer_id
group by c.customer_state
order by total_freight desc
```

Output:



Row	customer_state	total_freight	avg_freight
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:

By analysing the Query result We can find the state called “SP” has the highest total freight costs which could point to regions with higher shipping prices.Understanding the differences in order freight rates between states can offer information about local shipping habits, supplier locations, or client preferences that can be used to optimize processes and cut costs.

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.

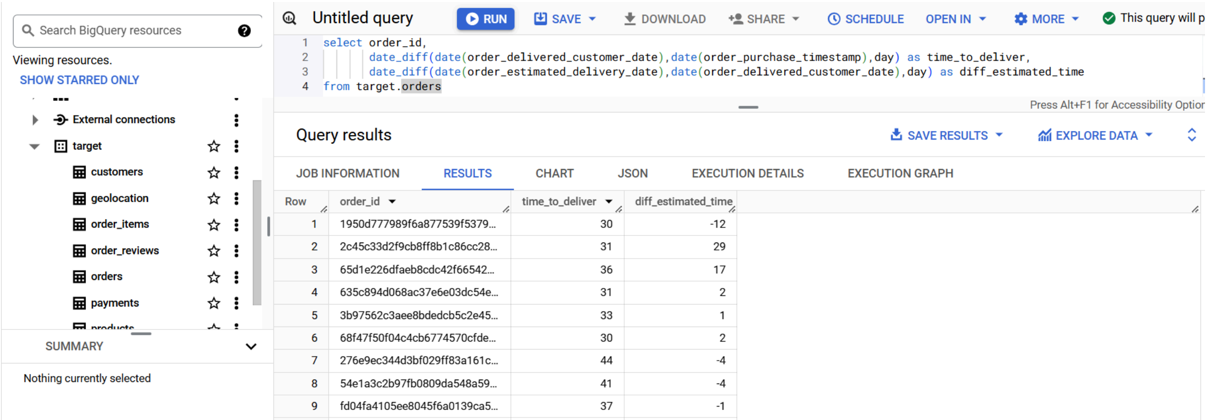
Do this in a single query.

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

```
select order_id,
       date_diff(date(order_delivered_customer_date),
            date(order_purchase_timestamp),day) as time_to_deliver,
       date_diff(date(order_estimated_delivery_date),
    date(order_delivered_customer_date),day) as diff_estimated_time
from target.orders
```

<u>Output:</u>



<u>Insights:</u>

By analysing the above query result we can find out the effectiveness of the delivery process, including any delays or early deliveries.It can be applied to manage customer expectations, enhance customer satisfaction, optimize the delivery process.

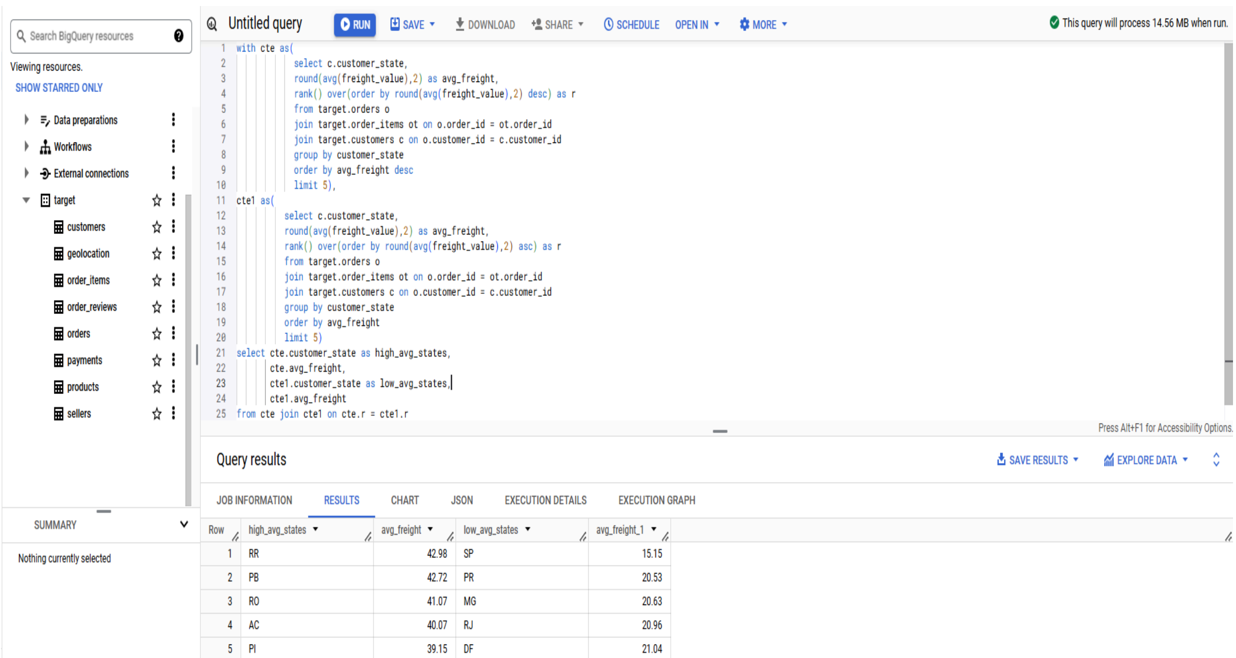
2. Find out the top 5 states with the highest & lowest average freight value.

```
with cte as(
select  c.customer_state,
        round(avg(freight_value),2) as avg_freight,
        rank() over(order by round(avg(freight_value),2) desc) as r
from target.orders o
join target.order_items ot on o.order_id = ot.order_id
join target.customers c on o.customer_id = c.customer_id
group by customer_state
order by avg_freight desc
limit 5),
```

```
cte1 as(
 select  c.customer_state,
         round(avg(freight_value),2) as avg_freight,
         rank() over(order by round(avg(freight_value),2) asc) as r
  from target.orders o
  join target.order_items ot on o.order_id = ot.order_id
  join target.customers c on o.customer_id = c.customer_id
  group by customer_state
  order by avg_freight
  limit 5)

select cte.customer_state as high_avg_states,
       cte.avg_freight,
       cte1.customer_state as low_avg_states,
       cte1.avg_freight
from cte join cte1 on cte.r = cte1.r
```

Output:



Insights:

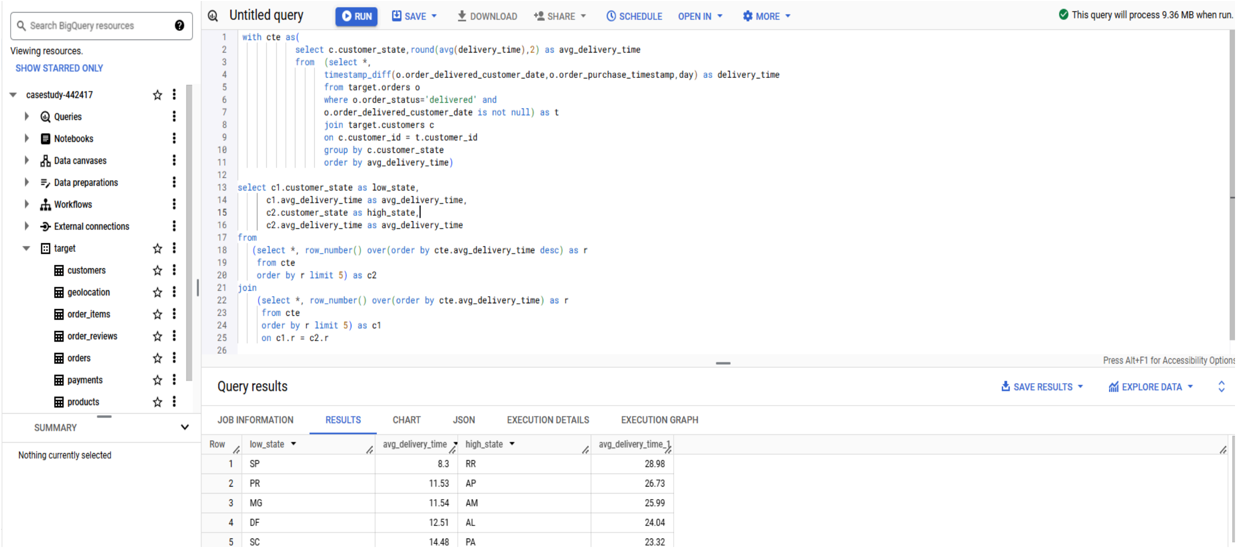
In the above query result we can find that The states with the highest average freight values like states called RR and PB may experience greater shipping prices due to various reasons.And the states with the lowest average freight values like states such as SP and PR relatively reduced shipping prices by looking at the average freight values.

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

```
with cte as(
  select c.customer_state,round(avg(delivery_time),2) as avg_delivery_time
  from  (select *,timestamp_diff(o.order_delivered_customer_date,
                              o.order_purchase_timestamp,day) as delivery_time
         from target.orders o
         where o.order_status='delivered' and
         o.order_delivered_customer_date is not null) as  t
         join target.customers c
         on c.customer_id = t.customer_id
         group by c.customer_state
        order by avg_delivery_time)

  select c1.customer_state as low_state,
        c1.avg_delivery_time as avg_delivery_time,
        c2.customer_state as high_state,
        c2.avg_delivery_time as avg_delivery_time
  from
   (select *, row_number() over(order by cte.avg_delivery_time desc) as r
    from cte
    order by r limit 5) as c2
    join
    (select *, row_number() over(order by cte.avg_delivery_time) as r
     from cte
     order by r limit 5) as c1
     on c1.r = c2.r
```

Output:



Insights:

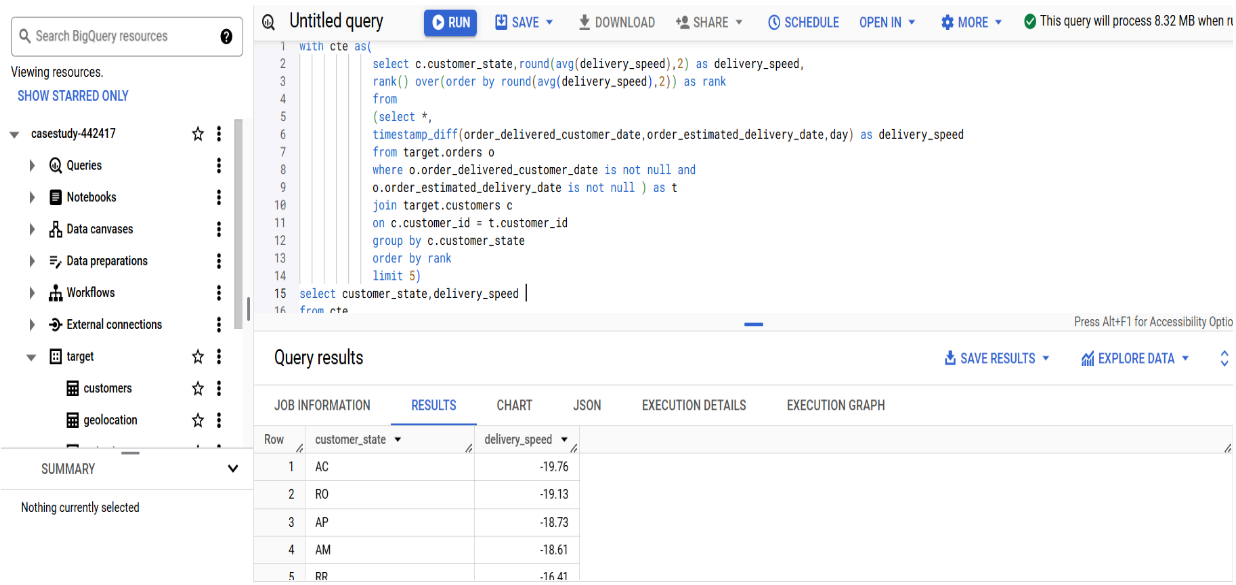
In the above query result we can find that the states with the highest average delivery time like states called RR and AP and the states with the lowest average delivery time like states called SP and PR.These insights can be helpful for our company looking to improve customer satisfaction, operational efficiency, delivery process optimization, and setting reasonable expectations for customers based on regional delivery time.

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.

```
with cte as
 (select c.customer_state,round(avg(delivery_speed),2) as delivery_speed,
        rank() over(order by round(avg(delivery_speed),2)) as rank
   from
   (select *,timestamp_diff(order_delivered_customer_date,
                        order_estimated_delivery_date,day) as delivery_speed
    from target.orders o
    where o.order_delivered_customer_date is not null and
    o.order_estimated_delivery_date is not null ) as t
    join target.customers c
    on c.customer_id = t.customer_id
    group by c.customer_state
    order by rank
    limit 5)

 select customer_state,delivery_speed
 from cte
```

Output:



Insights:

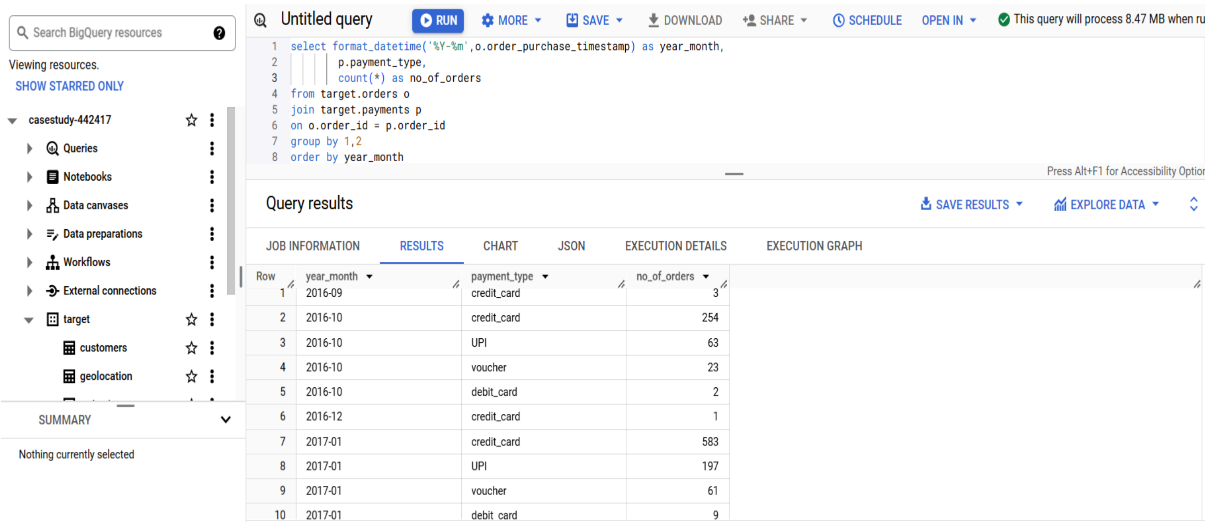
In the above query result we can find that AC,RO,AP,AM and RR are the top 5 states where the order delivery is really fast as compared to the estimated date of delivery.we can use this data to get more revenue from the customers of these states through increasing their orders.

Q6. Analysis based on the payments:

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

```
select format_datetime('%Y-%m',o.order_purchase_timestamp) as
        year_month,
        p.payment_type,
        count(*) as no_of_orders
 from target.orders o
join target.payments p
on o.order_id = p.order_id
group by 1,2
order by year_month
```

Output:



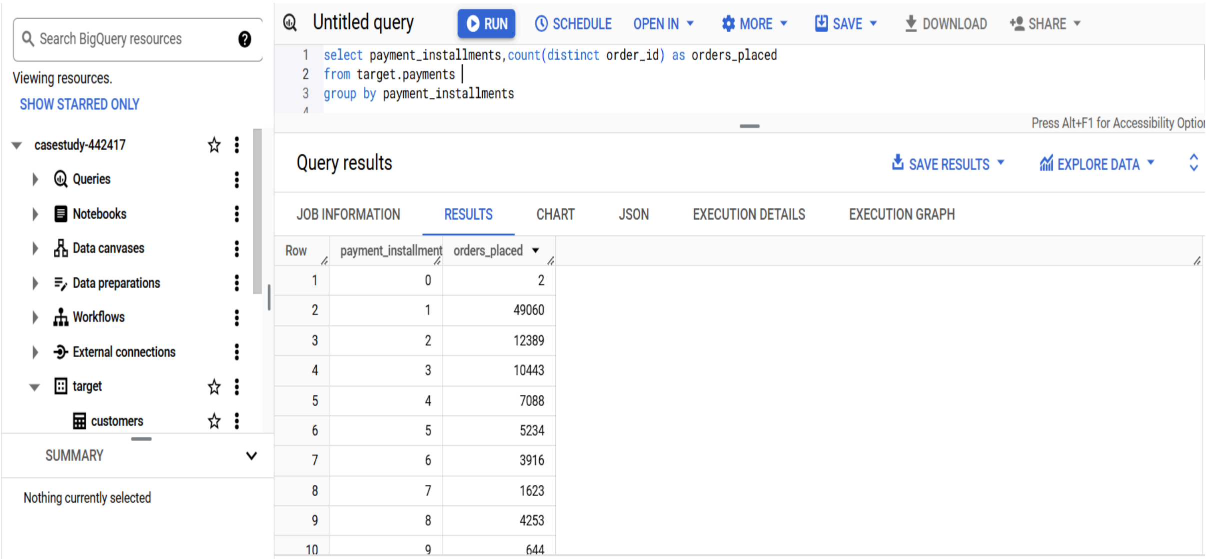
Insights:

In the above query result we can find that credit card as a payment method was most used in November 2017.Because highest number of orders placed in the november 2017 through credit card.

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

```
select payment_installments,
        count(distinct order_id) as orders_placed
from target.payments
group by payment_installments
```

Output:



Insights:

In the above query result we can find that 49060 orders were placed where payment instalment was 1. According to this analysis we can say that most of the customers are prefer
Or interested in the one payment instalment process.