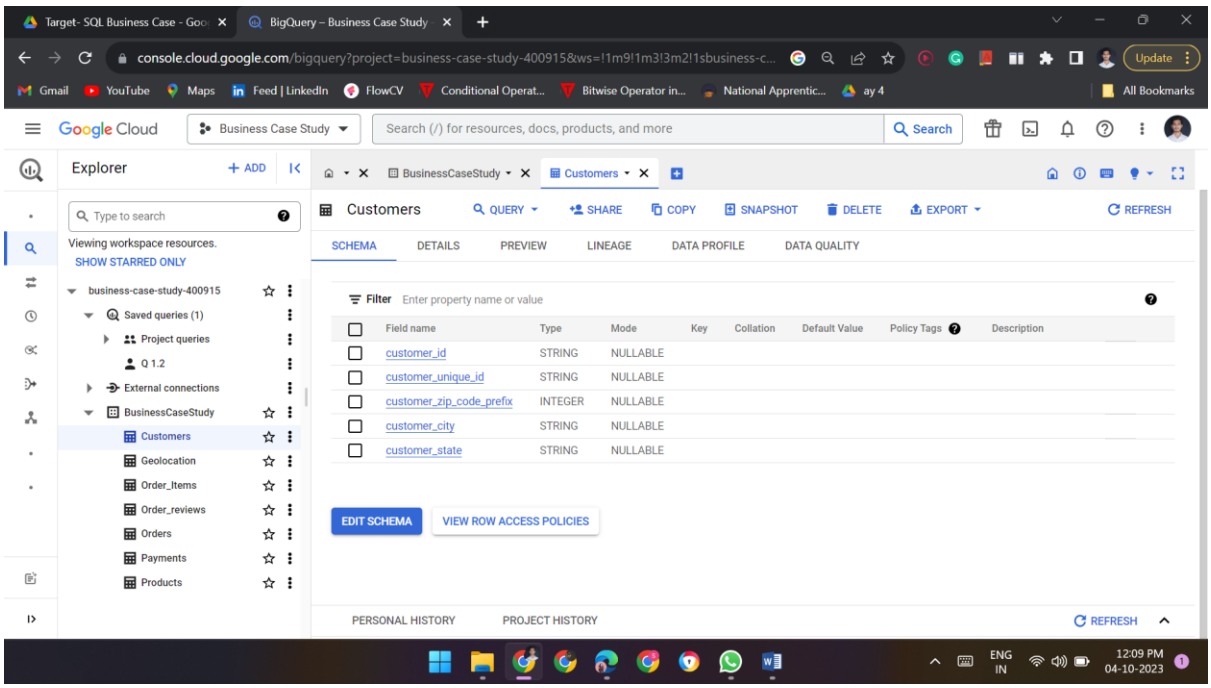
Business case study by Piyush Aswale

Case study topic: Target SQL

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

1.1Data type of all columns in the "customers" table. Solution: We can get to know the data type by clicking on the table name whose columns data type we want to know in bigquery.

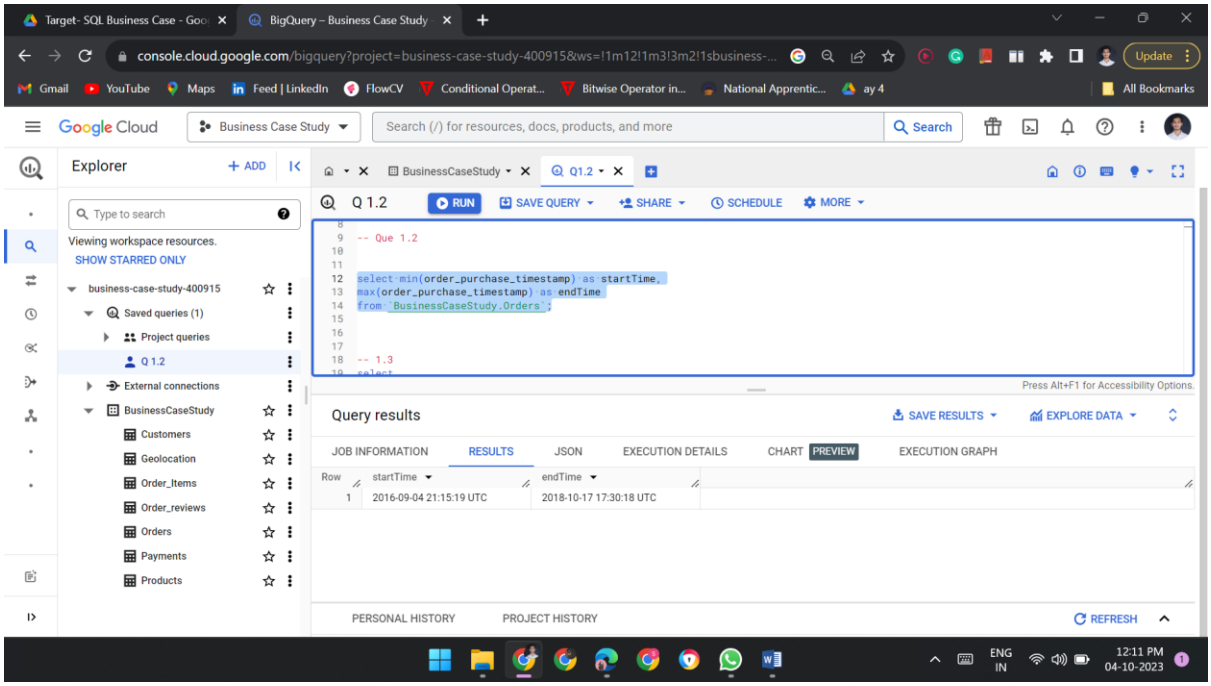


1.2.Get the time range between which the orders were placed.

Solution:

```
select min(order_purchase_timestamp) as startTime,
max(order_purchase_timestamp) as endTime
from `BusinessCaseStudy.Orders`;
```

The time range between which the orders we placed is from **2016-09-04 21:15:19 UTC to 2018-10-17 17:30:18 UTC**



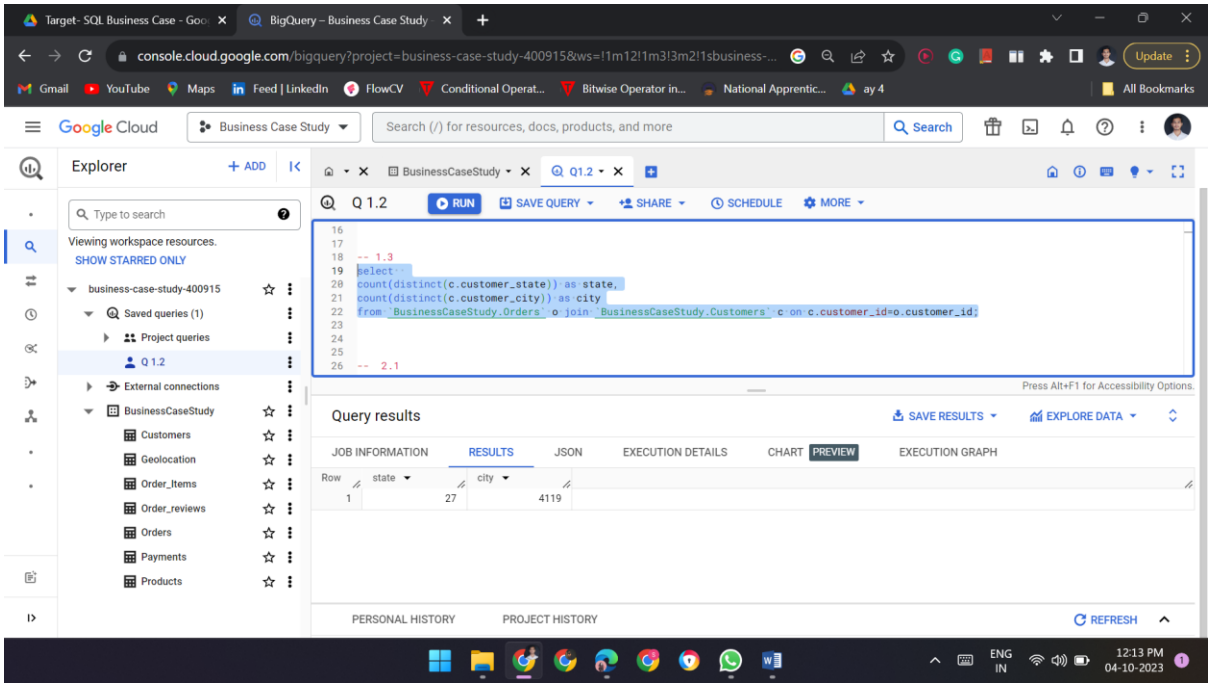
1.3Count the Cities & States of customers who ordered during the given period.

Solution:

Query: `select`

```
count(distinct(c.customer_state)) as state,
count(distinct(c.customer_city)) as city
from `BusinessCaseStudy.Orders` o join `BusinessCaseStudy.Customers` c on
c.customer_id=o.customer_id;
```

There were orders from 4119 cities and 27 states

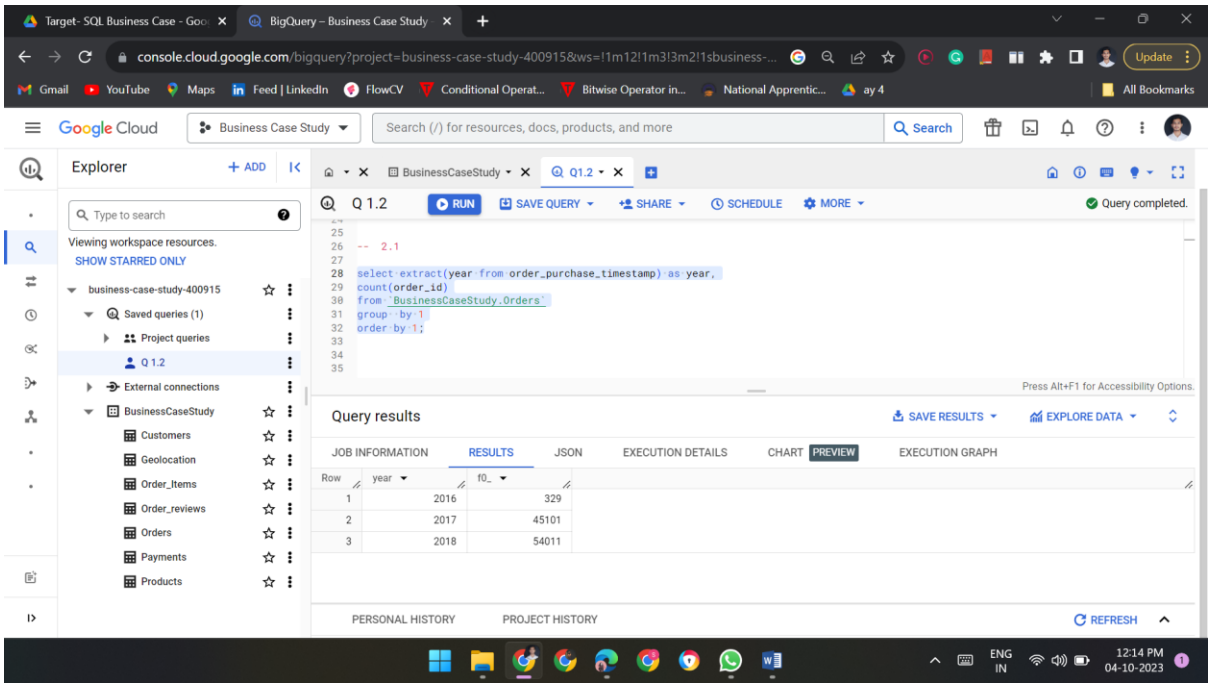


2 In-depth Exploration: 2.1 Is there a growing trend in the no. of orders placed over the past years? Solution:

Query:

```
select extract(year from order_purchase_timestamp) as year,

count(order_id)
from `BusinessCaseStudy.Orders`
group  by 1
order by 1;
```

Explanation: As per the given data, there is only 3 months data from year 2016, 12 months data from year 2017 and 10 month data from 2018, There is a good amount of growth in the numbers of orders placed from year 2016 to 2017, and a significant growth from 2017 to 2018



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

Query: select extract(year from order_purchase_timestamp) as year,

extract(month from order_purchase_timestamp) as month,

count(order_id) as Number_of_orders

from `target.orders` group by 1,2 order by 1,2

2.3 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

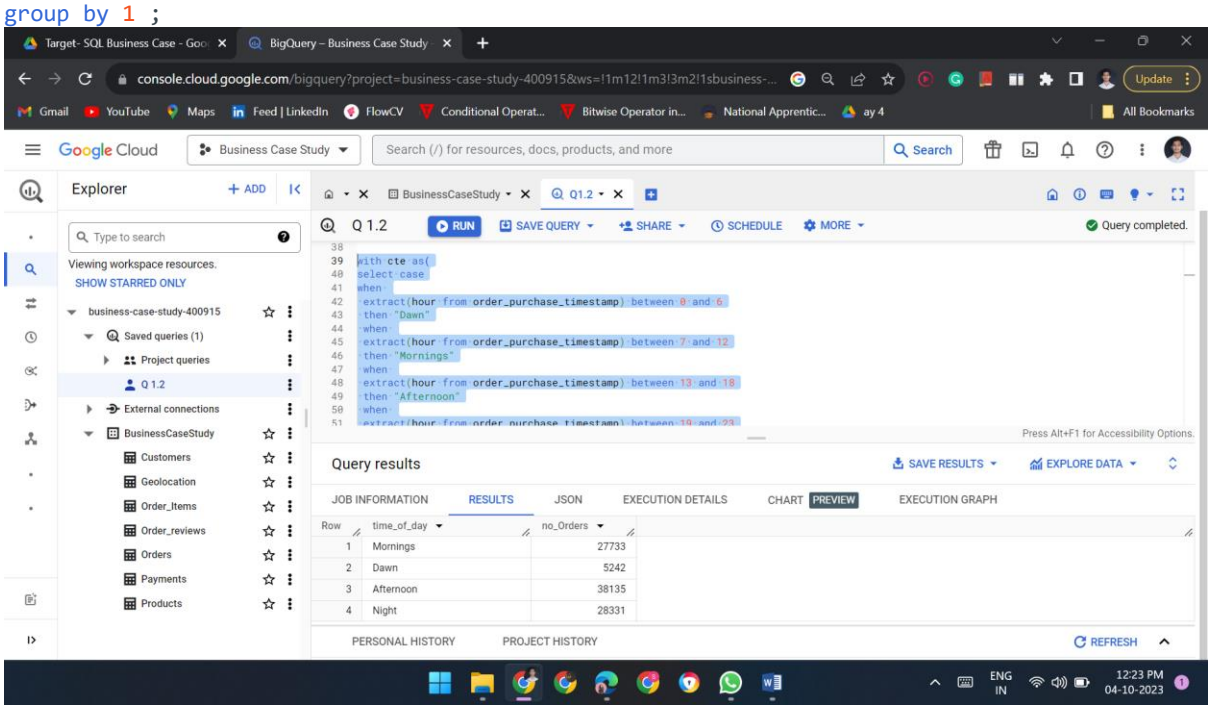
♣ 13-18 hrs : Afternoon

♣ 19-23 hrs : Night

Solution:

Query: `with cte as(`

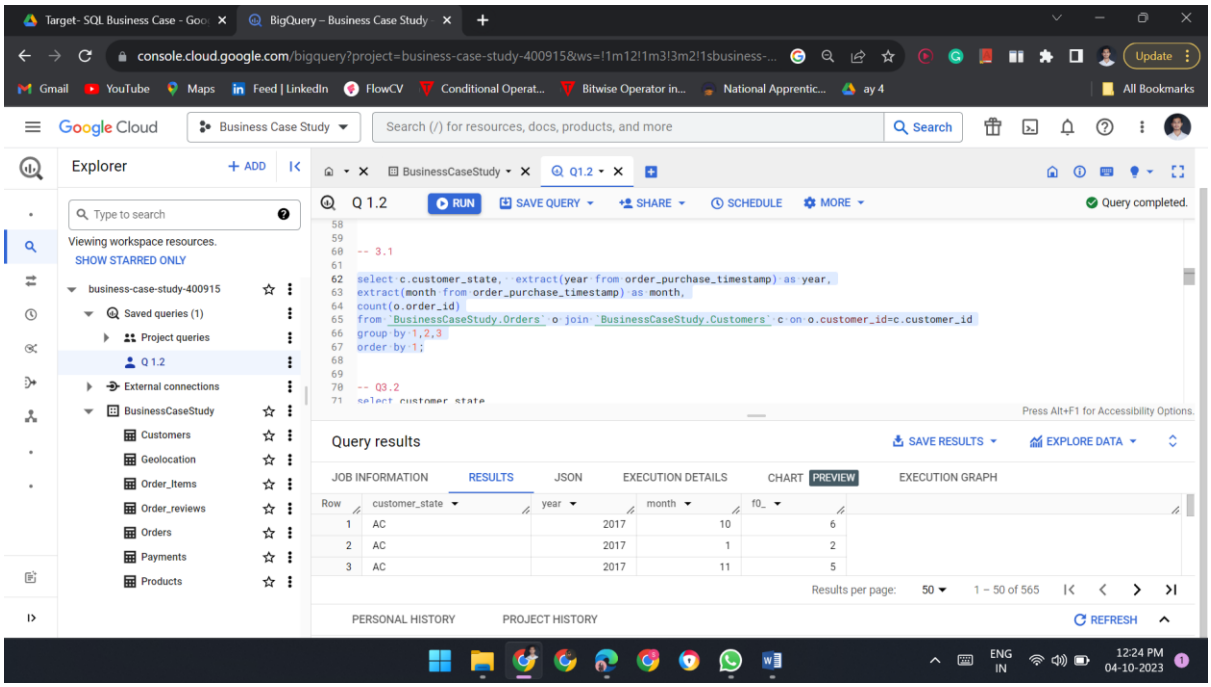
```
select case
when
 extract(hour from order_purchase_timestamp) between 0 and 6
 then "Dawn"
 when
 extract(hour from order_purchase_timestamp) between 7 and 12
 then "Mornings"
 when
 extract(hour from order_purchase_timestamp) between 13 and 18
 then "Afternoon"
 when
 extract(hour from order_purchase_timestamp) between 19 and 23
 then "Night"
end as time_of_Day
from `BusinessCaseStudy.Orders`)
select time_of_day, count(*) as no_Orders from cte
group by 1 ;
```



Evolution of E-commerce orders in the Brazil region: 3.1Get the month on month no. of orders placed in each state.

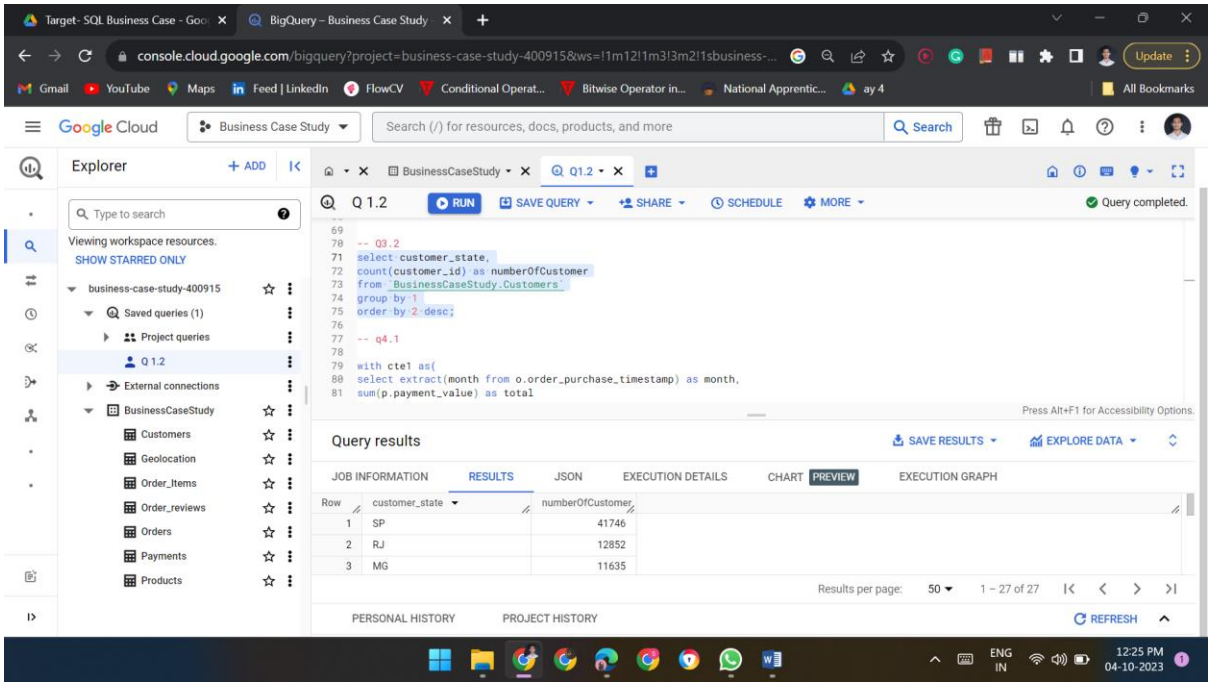
Solution:

Query:
```
select c.customer_state,  extract(year from order_purchase_timestamp) as year,

extract(month from order_purchase_timestamp) as month,
count(o.order_id)
from `BusinessCaseStudy.Orders` o join `BusinessCaseStudy.Customers` c on
o.customer_id=c.customer_id
group by 1,2,3
order by 1;
```
Explanation: Month on month orders placed data is as shown in the screenshot



3.2 How are the customers distributed across all the states?

Query:
```
select customer_state,

count(customer_id) as numberOfCustomer
from `BusinessCaseStudy.Customers`
group by 1
order by 2 desc;
```
Explanation: Distribution of the customers can be fetched using the above query with maximum customers from state SP with 41746 customer And least number of customers are from RR with only 46 customer



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

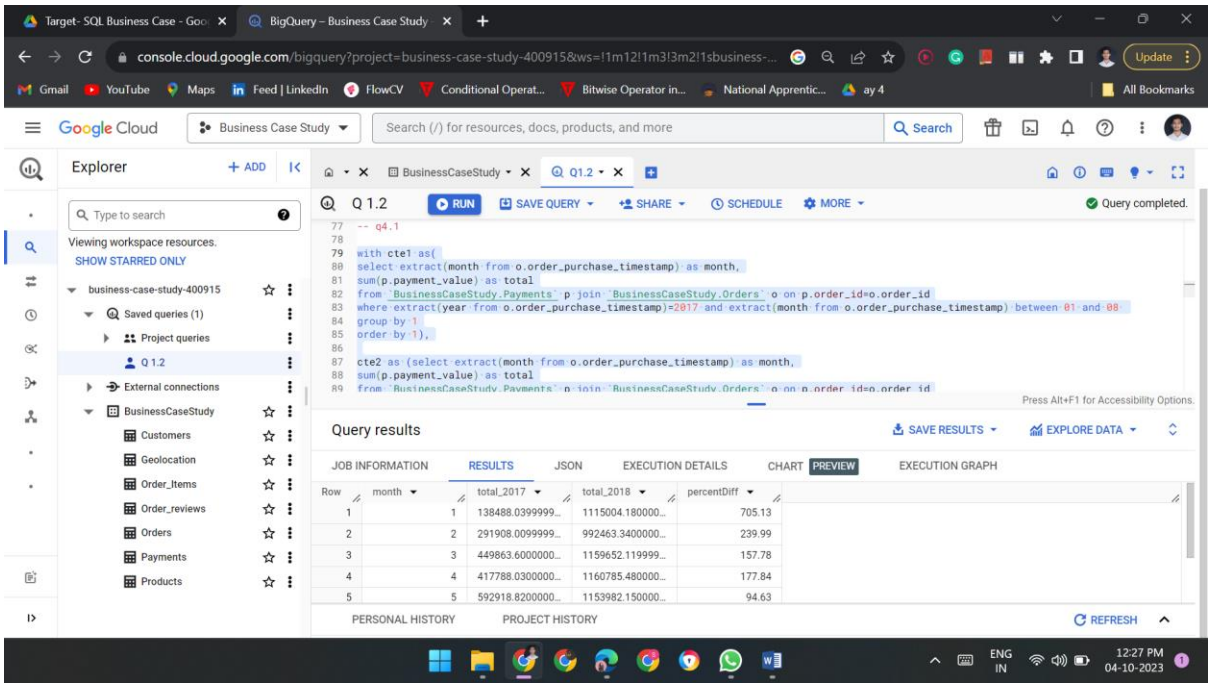
4.1 Get the % increase in the cost of orders from year 2017 to 2018 (include months between Jan to Aug only). You can use the "payment_value" column in the payments table to get the cost of orders.

Solution:

Query: `with cte1 as(`

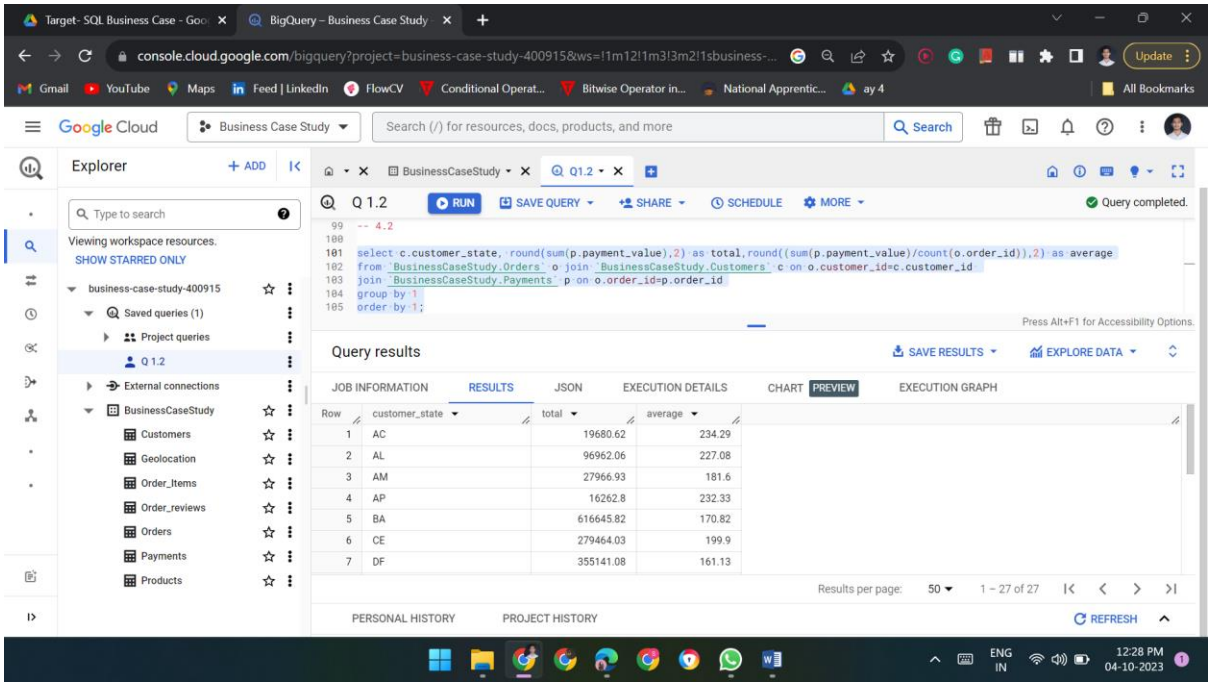
```
select extract(month from o.order_purchase_timestamp) as month,
sum(p.payment_value) as total
from `BusinessCaseStudy.Payments` p join `BusinessCaseStudy.Orders` o on
p.order_id=o.order_id
where extract(year from o.order_purchase_timestamp)=2017 and extract(month from
o.order_purchase_timestamp) between 01 and 08
group by 1
order by 1),

cte2 as (select extract(month from o.order_purchase_timestamp) as month,
sum(p.payment_value) as total
from `BusinessCaseStudy.Payments` p join `BusinessCaseStudy.Orders` o on
p.order_id=o.order_id
where extract(year from o.order_purchase_timestamp)=2018 and extract(month from
o.order_purchase_timestamp) between 01 and 08
group by 1
order by 1)

select c1.month,
c1.total as total_2017, c2.total as total_2018, round(((c2.total-c1.total)/c1.total)*100,2)
as percentDiff
from cte1 c1 join cte2 c2 on c1.month=c2.month
order by 1;
```

Explanation: There is a good percentage of growth in the cost of orders from year 2017 to 2018 monthly, with highest in January month 700%



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

Query:
```
select c.customer_state, round(sum(p.payment_value),2) as
total,round((sum(p.payment_value)/count(o.order_id)),2) as average

from `BusinessCaseStudy.Orders` o join `BusinessCaseStudy.Customers` c on
o.customer_id=c.customer_id
join `BusinessCaseStudy.Payments` p on o.order_id=p.order_id
group by 1
order by 1;
```
Explanation: The total and average value of order price of each state can be fetched using above query

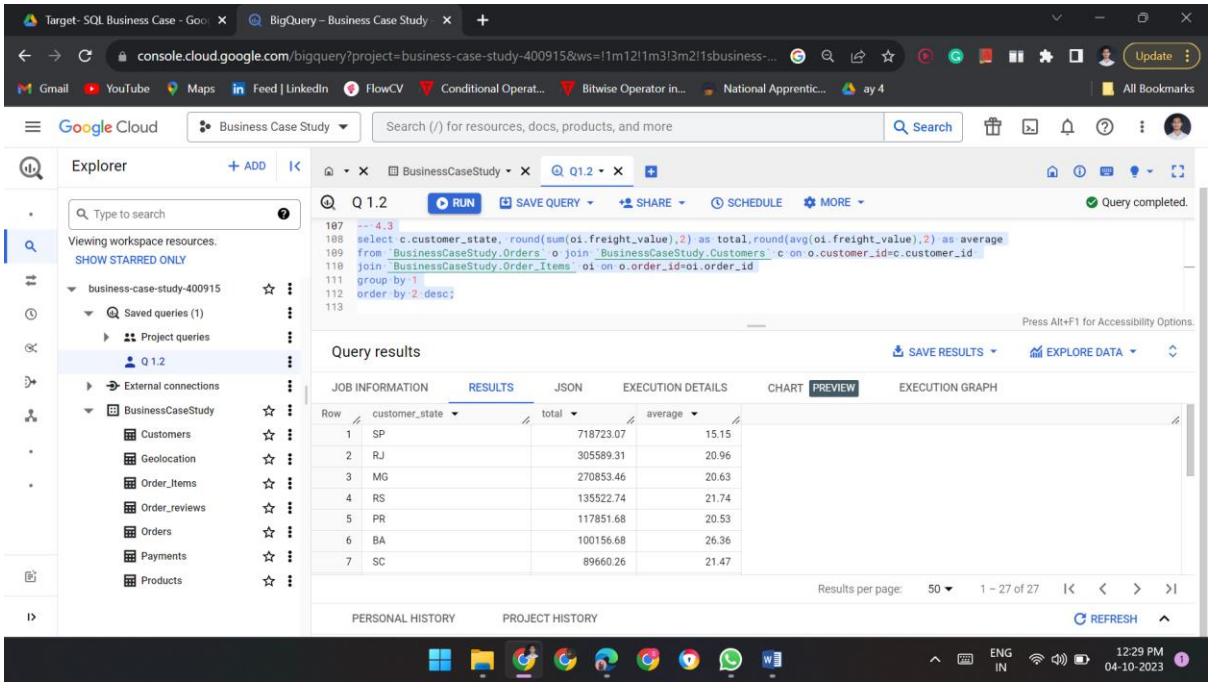


4.3 Calculate the Total & Average value of order freight for each state.

Query: -- 4.3

```
select c.customer_state, round(sum(oi.freight_value),2) as
total,round(avg(oi.freight_value),2) as average
from `BusinessCaseStudy.Orders` o join `BusinessCaseStudy.Customers` c on
o.customer_id=c.customer_id
join `BusinessCaseStudy.Order_Items` oi on o.order_id=oi.order_id
group by 1
order by 2 desc;
```

Explanation: The total and average value of order freight of each state can be fetched using above query.



Analysis based on sales, freight and delivery time.

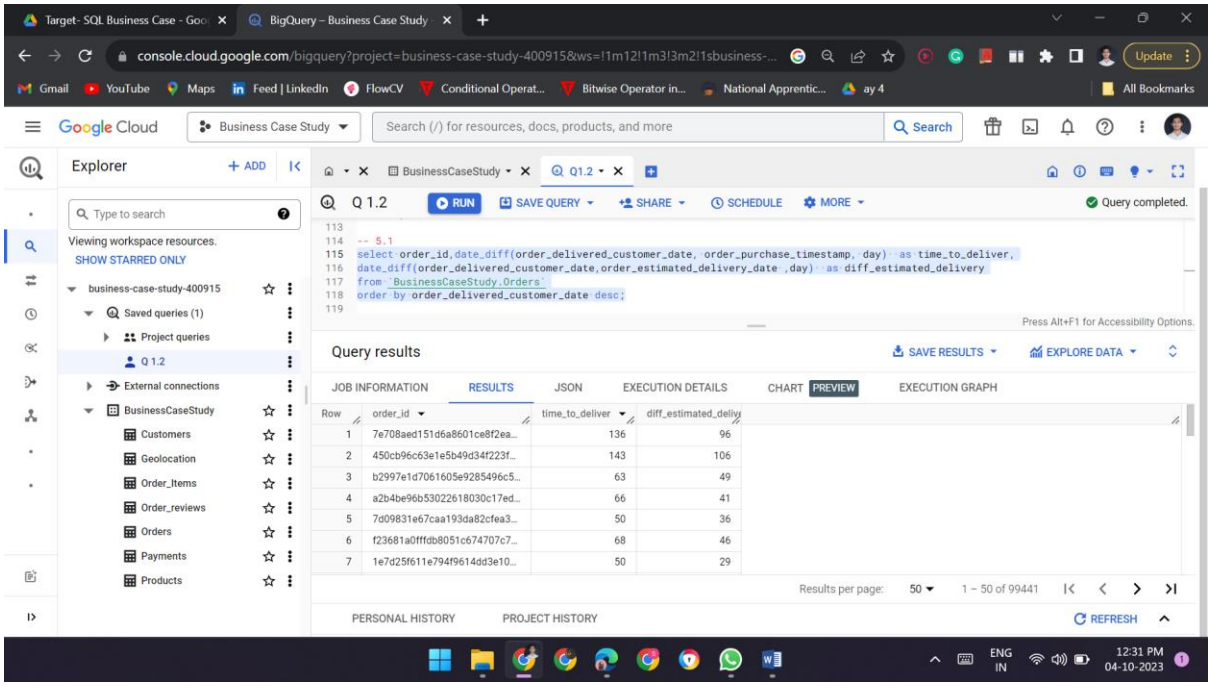
5.1 Find the no. of days taken to deliver each order from the order's purchase date as delivery time. Also, calculate the difference (in days) between the estimated & actual delivery date of an order. Do this in a single query. You can calculate the delivery time and the difference between the estimated & actual delivery date using the given formula:

i. **time_to_deliver** = order_delivered_customer_date - order_purchase_timestamp

ii. **diff_estimated_delivery** = order_estimated_delivery_date - order_delivered_customer_date

Query:
```
select order_id,date_diff(order_delivered_customer_date, order_purchase_timestamp, day)  as time_to_deliver,

date_diff(order_delivered_customer_date,order_estimated_delivery_date ,day)  as diff_estimated_delivery
from `BusinessCaseStudy.Orders`
order by order_delivered_customer_date desc;
```

Explanation: The required data about no. of days taken to deliver each order from the order's purchase date and the difference (in days) between the estimated & actual delivery date of an order is as shown below



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

Solution: Query for Highest value:

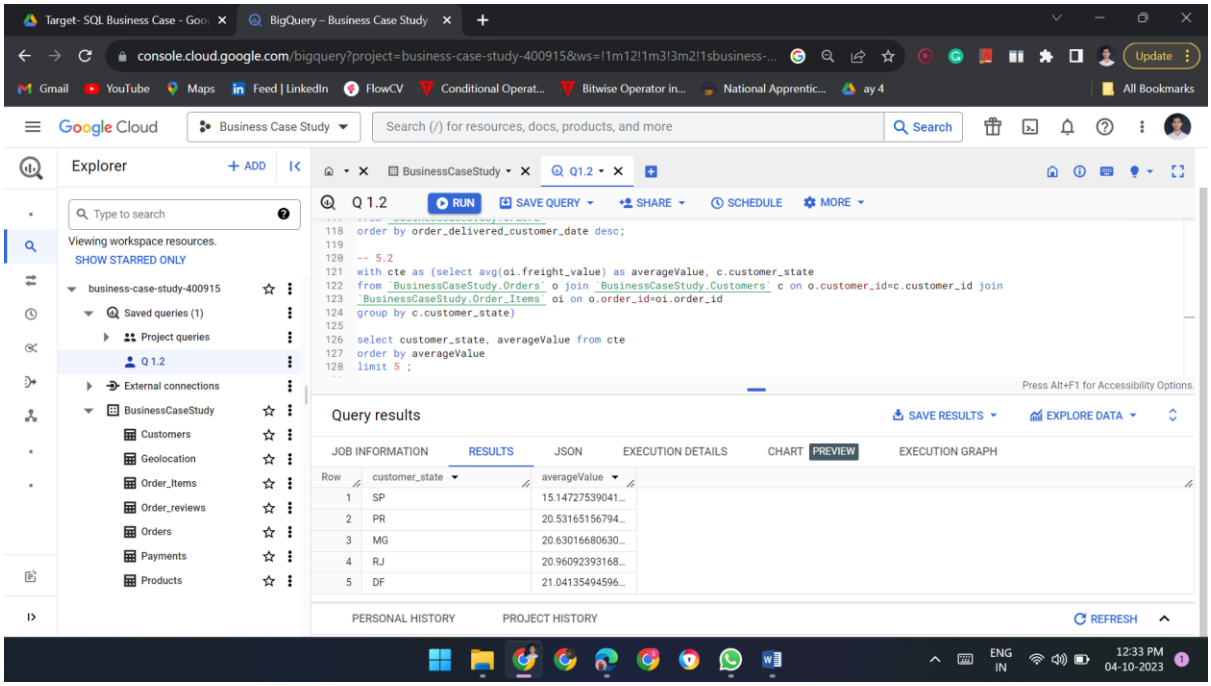
```
with cte as (select avg(oi.freight_value) as averageValue, c.customer_state
from `BusinessCaseStudy.Orders` o join `BusinessCaseStudy.Customers` c on
o.customer_id=c.customer_id join
`BusinessCaseStudy.Order_Items` oi on o.order_id=oi.order_id
group by c.customer_state)

select customer_state, averageValue from cte
order by averageValue
limit 5 ;
```

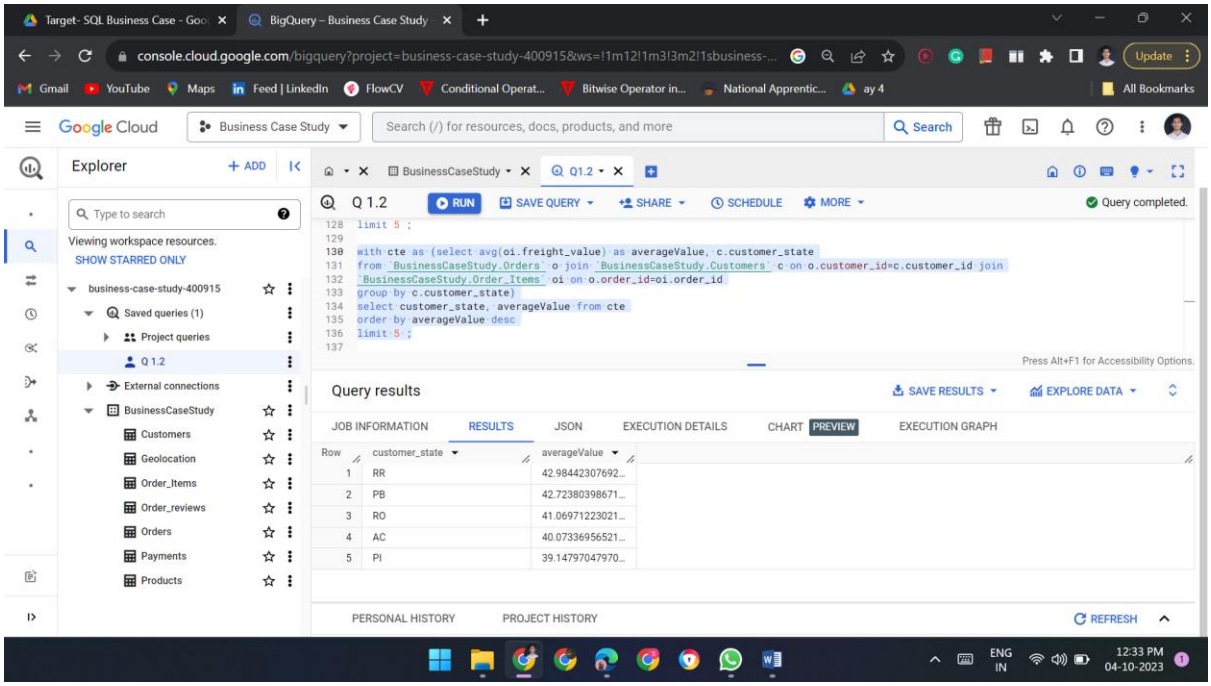
Explanation: The top 5 states with highest freight value are RR,PB,RO,AC,PI respectively

Solution: Query for Lowest value:

```
with cte as (select avg(oi.freight_value) as averageValue, c.customer_state
from `BusinessCaseStudy.Orders` o join `BusinessCaseStudy.Customers` c on
o.customer_id=c.customer_id join
`BusinessCaseStudy.Order_Items` oi on o.order_id=oi.order_id
group by c.customer_state)
select customer_state, averageValue from cte
order by averageValue desc
limit 5 ;
```

Explanation: The lowest 5 states with least freight value are SP,PR,MG,RJ,DF respectively





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

Solution: Query for highest delivery time:

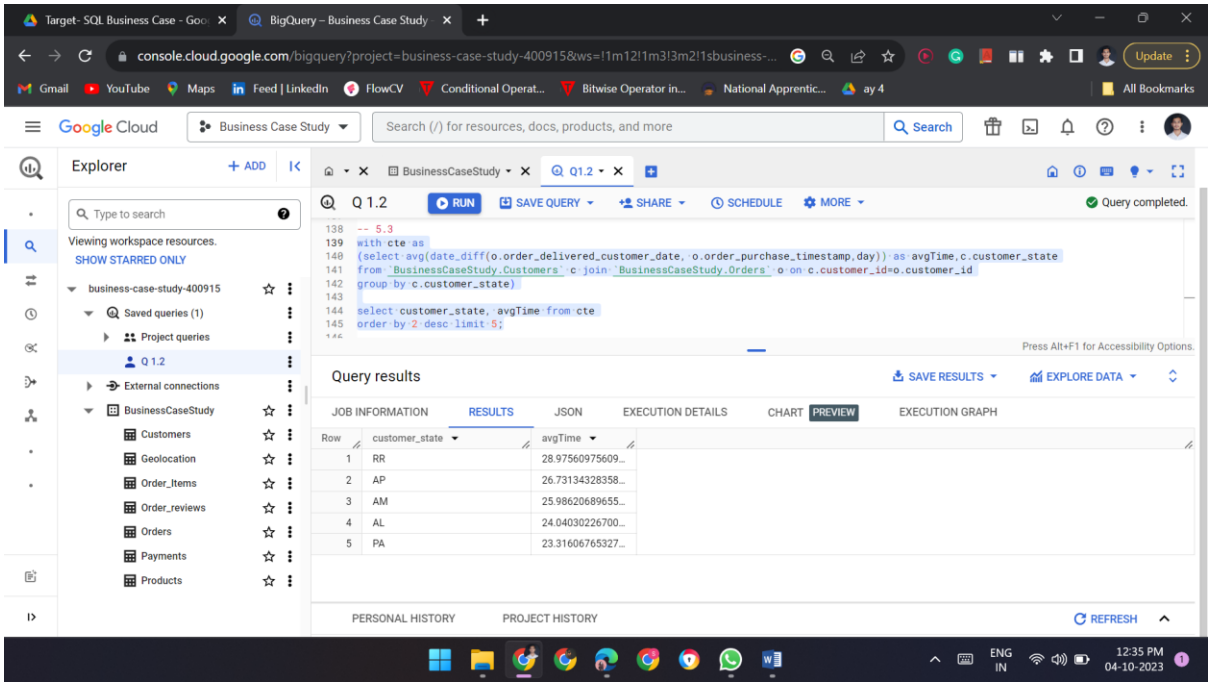
```
with cte as
(select avg(date_diff(o.order_delivered_customer_date, o.order_purchase_timestamp,day)) as avgTime,c.customer_state
from `BusinessCaseStudy.Customers` c join `BusinessCaseStudy.Orders` o on c.customer_id=o.customer_id
group by c.customer_state)
select customer_state, avgTime from cte
order by 2 desc limit 5;
```

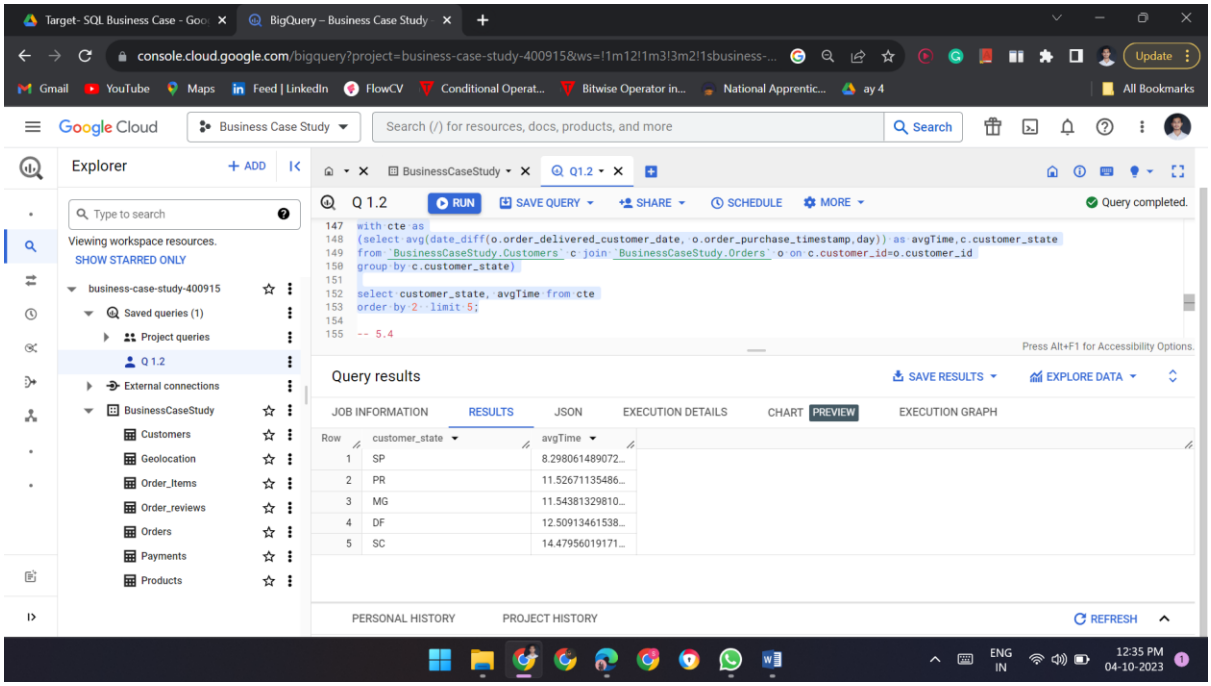
Explanation: Top 5 states with highest average delivery time are RR,AP,AM,AL,PA respectively.

Query for least delivery time:

```
with cte as
(select avg(date_diff(o.order_delivered_customer_date, o.order_purchase_timestamp,day)) as
avgTime,c.customer_state
from `BusinessCaseStudy.Customers` c join `BusinessCaseStudy.Orders` o on
c.customer_id=o.customer_id
group by c.customer_state)

select customer_state, avgTime from cte
order by 2  limit 5;
```

Explanation: Top 5 states with least average delivery time are SP,PR,MG,DF,SC respectively



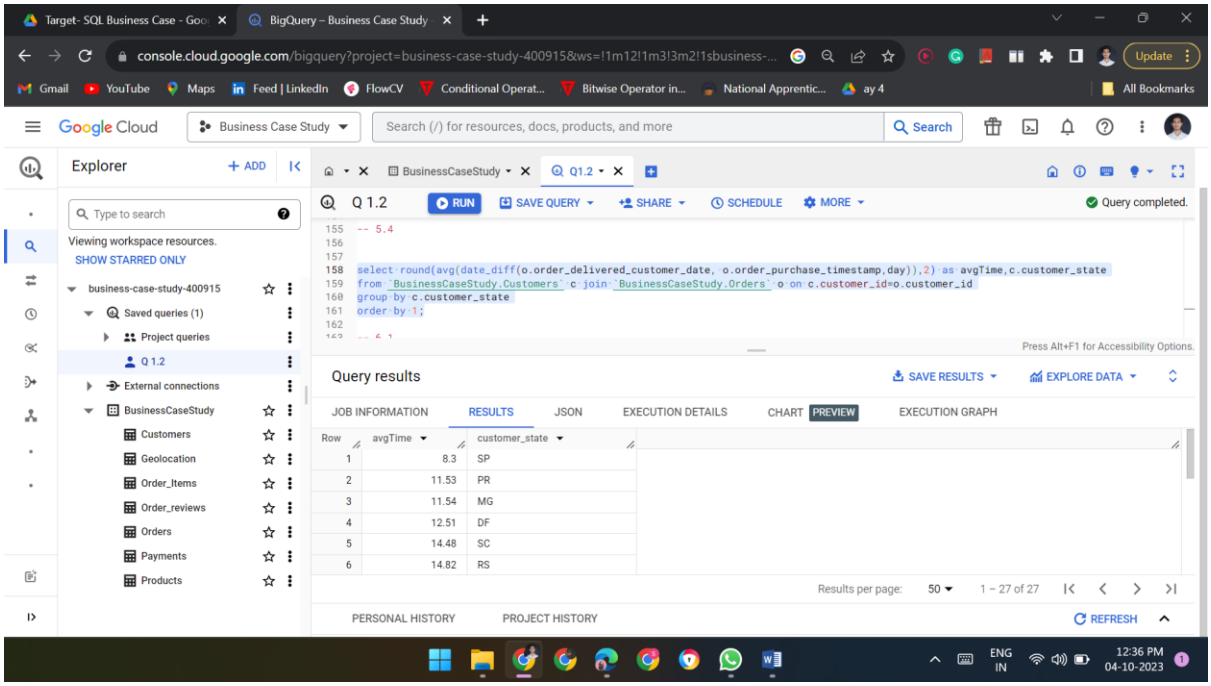


5.4 Find out the top 5 states where the order delivery is really fast as compared to the estimated date of delivery. You can use the difference between the averages of actual & estimated delivery date to figure out how fast the delivery was for each state.

Solution: Query:

```
select round(avg(date_diff(o.order_delivered_customer_date,
o.order_purchase_timestamp,day)),2) as avgTime,c.customer_state
from `BusinessCaseStudy.Customers` c join `BusinessCaseStudy.Orders` o on
c.customer_id=o.customer_id
group by c.customer_state
order by 1;
```

Explanation: The top 5 states where the order delivery is really fast as compared to estimated date of delivery are AL,MA,SE,ES,BA

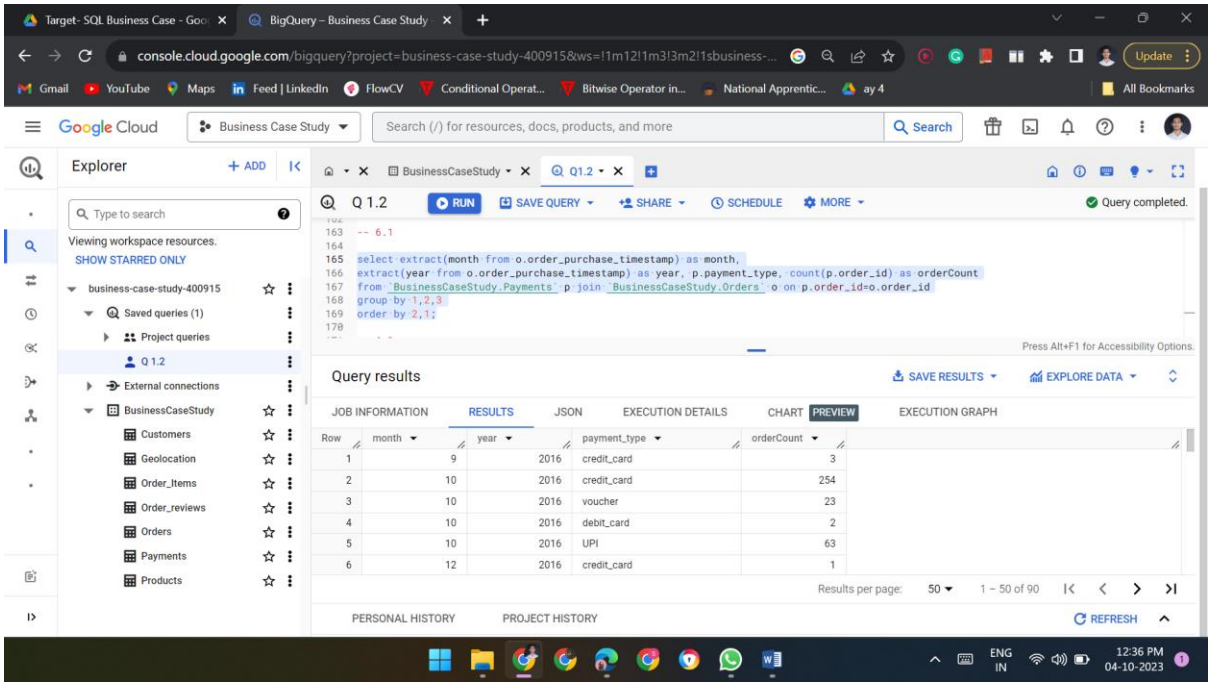


6. Analysis based on the payments: 6.1 Find the month on month no. of orders placed using different payment types.

Solution: Query:

```
select extract(month from o.order_purchase_timestamp) as month,
extract(year from o.order_purchase_timestamp) as year, p.payment_type, count(p.order_id) as
orderCount
from `BusinessCaseStudy.Payments` p join `BusinessCaseStudy.Orders` o on
p.order_id=o.order_id
group by 1,2,3
order by 2,1;
```

Explanation: The above query gives us the month on month no. of orders placed using different payments type



6.2Find the no. of orders placed on the basis of the payment installments that have been paid.

Solution: Query: `select p.payment_installments, count(o.order_id) as numberOfOrder`

```
from `BusinessCaseStudy.Payments` p join `BusinessCaseStudy.Orders` o on
p.order_id=o.order_id
where p.payment_installments != 0
group by 1
```

The above query gives us no. of orders placed on the basis of the payment installments that have been paid.